KRYLOV SUBSPACE RECYCLING FOR SEQUENCES OF SHIFTED LINEAR SYSTEMS*

KIRK M. SOODHALTER[†], DANIEL B. SZYLD[‡], AND FEI XUE[§]

Abstract. Subspace recycling methods, a class of Krylov subspace deflation techniques, have been shown to have the potential to accelerate convergence of Krylov subspace methods. In particular, they can be useful when solving sequences of slowly-changing linear systems. We wish to extend such methods to solve sequences of linear systems, where for each system, we also solve a family of shifted systems in which the coefficient matrices only differ by multiples of the identity from a base system matrix. In this work, we demonstrate the difficulty of extending recycling techniques to solve multiple shifted systems while maintaining the fixed storage property. As an alternative, we introduce a scheme which constructs approximate corrections to the solutions of the shifted systems at each cycle while only minimizing the base system residual. At convergence of the base system solution, we apply the method recursively to the remaining unconverged systems. The method is robust enough to be applied to sequences of systems where the base system changes slowly and the shifts differ for each base system. We present numerical examples involving systems arising in lattice quantum chromodynamics.

Key words. Krylov subspace methods, subspace recycling, shifted linear systems, QCD

1. Introduction.

We consider the solution of a sequence of families of non-Hermitian linear systems. Let $\mathcal{F}$ denote a family of coefficient matrices differing by multiples of the identity. In other words,

$$\mathcal{F} = \left\{ \mathbf{A} + \sigma^{(\ell)}\mathbf{I} \right\}_{\ell=1}^{L} \subset \mathbb{C}^{n\times n}, \tag{1.1}$$

where L is the number of matrices in the family, and we are solving the family of linear systems

$$\left(\mathbf{A} + \sigma^{(\ell)}\mathbf{I}\right)\mathbf{x}^{(\ell)} = \mathbf{b} \quad \text{for} \quad \ell = 1, \ldots, L. \tag{1.2}$$

We call the numbers $\left\{\sigma^{(\ell)}\right\}_{\ell=1}^{L} \subset \mathbb{C}$ *shifts*, $\mathbf{A}$ the *base matrix*, and $\mathbf{A} + \sigma\mathbf{I}$ a *shifted matrix*. Systems of the form (1.2) are called *shifted linear systems*. There are many applications which warrant the solution of a family of shifted linear systems with coefficient matrices belonging to $\mathcal{F}$, such as those arising in lattice quantum chromodynamics (QCD) (see, e.g., [15]) as well as other applications such as Tikhonov-Philips regularization, global methods of nonlinear analysis, and Newton trust region methods [5]. Krylov subspace methods have been proposed to simultaneously solve this family of systems [13, 14, 30]. Our goal is to adapt subspace recycling technology to accelerate existing methods for solving sequences of shifted linear systems. In this paper, we describe the challenges in designing a method which extends subspace

*This version dated January 16, 2013. This research was supported in part by the U.S. Department of Energy under grant DE-FG02-05ER25672 and the U.S. National Science Foundation under grant DMS-1115520.

[†]Industrial Mathematics Institute, Johannes Kepler University, Altenbergerstraße 69, A-4040 Linz, Austria. (soodhalter@indmath.uni-linz.ac.at)

[‡]Department of Mathematics, Temple University, 1805 N Broad Street, Philadelphia, PA 19122-6094. (szyld@temple.edu).

[§]Department of Mathematics, University of Louisiana at Lafayette, Lafayette, LA 70504. (fxue@louisiana.edu)

recycling technology [23] to this setting. We then present a strategy which mitigates these difficulties.

Let $\mathcal{F}_i$ denote the ith family of linear systems, defined by

$$\mathcal{F}_i = \left\{\mathbf{A}_i + \sigma_i^{(\ell)}\mathbf{I}\right\}_{\ell=1}^{L_i} \subset \mathbb{C}^{n\times n},$$

where L_i denotes the number of linear systems to be solved at step i. In other words, at step i, for shifts $\left\{\sigma_i^{(\ell)}\right\}_{\ell=1}^{L_i}$ we are solving systems of the form

$$\left(\mathbf{A}_i + \sigma_i^{(\ell)}\mathbf{I}\right)\mathbf{x}_i^{(\ell)} = \mathbf{b}_i \quad \text{for} \quad \ell = 1 \ldots L_i.$$

This is the problem we target in this paper. However, for simplicity of discussion, we frequently will focus on one family of linear systems with only one shift and drop the index i, yielding two systems of the form

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{1.3}$$

$$(\mathbf{A} + \sigma\mathbf{I})\mathbf{x}^{(\sigma)} = \mathbf{b}. \tag{1.4}$$

When dealing with only one family, in the absence of initial deflation subspace, techniques have been developed to solve the family of systems simultaneously, building recycled subspaces from harmonic Ritz vectors; see, e.g., [8].

In the next section, we review some existing methods for solving (1.3) and (1.4), and we describe the framework of subspace recycling used in, e.g., [23]. In Section 3, we discuss the difficulties of adapting subspace recycling to shifted systems. We introduce a new algorithm (ideal but memory-intensive and computationally expensive) which combines ideas for solving (1.3) and (1.4) simultaneously while using subspace recycling techniques. This method is a direct extension of the one presented in [14]. In Section 4, we show that it is *not* possible to construct solutions for all shifted systems over the same augmented subspace in a way that is compatible with restarting. We present a scheme in Section 5 which will produce improved approximations for the shifted system while solving the base system, using only one recycled subspace. This method is derived from the shifted GMRES method [14], but it is not a direct extension. In Section 6, we present numerical results for a family of simple bidiagonal matrices and for some sequences of QCD matrices obtained from [1] and [20].

2. Preliminaries. In many Krylov subspace iterative methods, recall that we generate an orthonormal basis for

$$\mathcal{K}_j(\mathbf{A}, \mathbf{u}) = \operatorname{span}\left\{\mathbf{u}, \mathbf{A}\mathbf{u}, \ldots, \mathbf{A}^{j-1}\mathbf{u}\right\} \tag{2.1}$$

with the Arnoldi process, where $\mathbf{u}$ is some starting vector. Let $\mathbf{V}_j \in \mathbb{C}^{n\times j}$ be the matrix with orthonormal columns generated by the Arnoldi process spanning $\mathcal{K}_j(\mathbf{A}, \mathbf{u})$. Then we have the Arnoldi relation

$$\mathbf{A}\mathbf{V}_j = \mathbf{V}_{j+1}\overline{\mathbf{H}}_j \tag{2.2}$$

with $\overline{\mathbf{H}}_j \in \mathbb{C}^{(j+1)\times j}$; see, e.g., [26, Section 6.3] and [32]. Let $\mathbf{x}_0$ be an initial approximation to the solution of a linear system we wish to solve. A Krylov subspace method for solving a linear system generates successively larger Krylov subspaces at each iteration; and at iteration j, the method generates an approximation of the form

$\mathbf{x}_j = \mathbf{x}_0 + \mathbf{t}_j$, where $\mathbf{t}_j \in \mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)$ and $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$. We choose $\mathbf{t}_j$ by requiring that the residual $\mathbf{r}_j = \mathbf{b} - \mathbf{A}\mathbf{x}_j$ satisfies some constraint. For example, solutions produced by GCR [11] and GMRES [27] satisfy $\mathbf{r}_j \perp \mathbf{A}\mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)$, which is equivalent to $\mathbf{t}_j$ solving the minimization problem

$$\mathbf{t}_j = \underset{\mathbf{t} \in \mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)}{\operatorname{argmin}} \left\| \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{t}) \right\|,$$

and this is equivalent to solving the smaller minimization problem

$$\mathbf{y}_j = \underset{\mathbf{y} \in \mathbb{C}^j}{\operatorname{argmin}} \left\| \overline{\mathbf{H}}_j \mathbf{y} - \left\| \mathbf{r}_0 \right\| \mathbf{e}_1^{(j+1)} \right\|,$$

where we use the notation $\mathbf{e}_j^{(i)}$ to denote the jth Cartesian basis vector in $\mathbb{R}^i$, and setting $\mathbf{x}_j = \mathbf{x}_0 + \mathbf{V}_j \mathbf{y}_j$.

For a Krylov subspace method, one can describe a restarted version of that method. The memory needed to store $\mathbf{V}_j$ for GMRES increases with j, and a restart is often used when the size of $\mathbf{V}_j$ exceeds the limit of available memory. Recall that in restarted GMRES, often called GMRES(m), we run an m-step cycle of the GMRES method and compute an approximation $\mathbf{x}_m$. We then discard all Arnoldi vectors and use $\mathbf{x}_m$ as the initial approximation for the next cycle of GMRES. This process is repeated until we achieve convergence. Adaptions of restarted GMRES to solve (1.3) and (1.4) have been previously proposed; see, e.g., [14].

It should be noted that methods based on the nonsymmetric Lanczos process have also been adapted for solving multiple shifted systems. Extensions of methods, such as BiCGStab [13] and QMR, have been developed [15]. A recently proposed method called IDR [35], which has been shown to be a generalization of BiCGStab [33], has also been extended to solve (1.3) and (1.4) [19]. We will not deal with nonsymmetric Lanczos-based methods in this paper, but these alternatives are worth mentioning.

Many methods for solving (1.3) and (1.4) use the fact that for any shift σ, the Krylov subspace generated by $\mathbf{A}$ and $\mathbf{b}$ is invariant under the shift, i.e.,

$$\mathcal{K}_j(\mathbf{A}, \mathbf{b}) = \mathcal{K}_j(\mathbf{A} + \sigma\mathbf{I}, \mathbf{b}),$$

as long as the starting vectors are collinear, with a shifted Arnoldi relation similar to (2.2)

$$(\mathbf{A} + \sigma\mathbf{I})\mathbf{V}_j = \mathbf{V}_{j+1} \overline{\mathbf{H}}_j^{(\sigma)}. \tag{2.3}$$

Note that the shift-invariance no longer holds if general preconditioning is used. However, polynomial preconditioning [16] would be appropriate in this setting. There has been recent work on choosing optimal polynomial preconditioners in the setting of solving multiple shifted systems [2]. In this project, though, we focus on the unpreconditioned case, as in [14] and [22].

The shift-invariance property indicates that large savings in storage and time can be achieved by generating only one sequence of Krylov subspaces and solving all shifted systems in one Krylov subspace simultaneously. Suppose that the initial residuals of (1.3) and (1.4) are collinear. For non-restarted Krylov subspace methods derived by applying a Petrov-Galerkin condition to the residual, it is straightforward to formulate a shifted version of that method. We simply apply the same Petrov-Galerkin condition for all residuals. However, once restarting is introduced, the situation becomes more complicated. If we project the residuals such that they are no longer collinear, at restart the Krylov subspace for the base system and that of the shifted systems are no longer identical. This loss of collinearity does not occur for methods such as restarted Full Orthogonalization Method (FOM) [30]. FOM is a method closely related to GMRES, and the residuals produced by FOM are all inherently collinear with the newest Arnoldi vector. In [13], a general theorem is presented which describes conditions under which the residuals will be naturally collinear in this manner. However, as discussed by Frommer and Glässner in [14], the GMRES residual projection does not have this property. We describe the algorithm proposed in [14] in more detail.

Frommer and Glässner [14] proposed a restarted GMRES method to solve (1.3) and (1.4). Suppose that for the base system and the shifted systems we have the same initial approximation $\mathbf{x}_0 = \mathbf{x}_0^{(\sigma)} = \mathbf{0}$, meaning that the residuals for the base and shifted systems are the same, i.e., $\mathbf{r}_0 = \mathbf{r}_0^{(\sigma)} = \mathbf{b}$. A restarted GMRES method to solve (1.3) and (1.4) will require that we enforce collinearity upon the residuals at the end of each cycle rather than simply minimizing all residuals.

Suppose that our residuals for the shifted and base system satisfies the relation

$$\mathbf{r}_0^{(\sigma)} = \beta_0 \mathbf{r}_0 \tag{2.4}$$

at the beginning of a cycle. For the base system, we can represent the GMRES minimized residual at the mth step as $\mathbf{r}_m = r_m(\mathbf{A})\mathbf{r}_0$ where the *residual polynomial* $r_m(t)$ is a polynomial of degree less than or equal to m such that $r_m(0) = 1$. We enforce the condition that the residual for the shifted system is parallel to the minimized residual of the base system, i.e.,

$$\mathbf{r}_m^{(\sigma)} = \beta_m \mathbf{r}_m. \tag{2.5}$$

By straightforward computations in [14], it is shown that if $\mathbf{z}_{m+1} = \|\mathbf{r}_0\| \mathbf{e}_1 - \overline{\mathbf{H}}_m \mathbf{y}_m$, then for (2.5) to hold, we must have

$$\overline{\mathbf{H}}_m^{(\sigma)} \mathbf{y}_m^{(\sigma)} + \mathbf{z}_{m+1}\beta_m = \beta_0 \|\mathbf{r}_0\| \mathbf{e}_1^{(m+1)}, \tag{2.6}$$

and $\mathbf{x}_m^{(\sigma)} = \mathbf{x}_0^{(\sigma)} + \mathbf{V}_m \mathbf{y}_m^{(\sigma)}$, where

$$\overline{\mathbf{H}}_m^{(\sigma)} = \overline{\mathbf{H}}_m + \begin{bmatrix} \sigma \mathbf{I}_{m\times m} \\ \mathbf{0}_{1\times m} \end{bmatrix}.$$

Thus, we can compute both $\mathbf{y}_m^{(\sigma)}$ and β_m by solving the augmented linear system,

$$\begin{bmatrix} \overline{\mathbf{H}}_m^{(\sigma)} & \mathbf{z}_{m+1} \end{bmatrix} \begin{bmatrix} \mathbf{y}_m^{(\sigma)} \\ \beta_m \end{bmatrix} = \beta_0 \|\mathbf{r}_0\| \mathbf{e}_1^{(m+1)}. \tag{2.7}$$

Note that we only compute the minimum residual solution using a Petrov-Galerkin condition for the base system. We enforce collinearity of the residuals to compute the approximation for the shifted system. We can equivalently think of this as applying an oblique rather than an orthogonal projection to the residual of the shifted system on $\mathbf{A}\mathcal{K}_m(\mathbf{A}, \mathbf{r}_0)$. It is shown in [14, Lemmas 2.1 and 2.4] that a solution to (2.7) exists if and only if the residual polynomial $r_m(t)$ satisfies $r_m(-\sigma) \neq 0$; otherwise, the augmented system is singular. If we compute the QR-factorization,

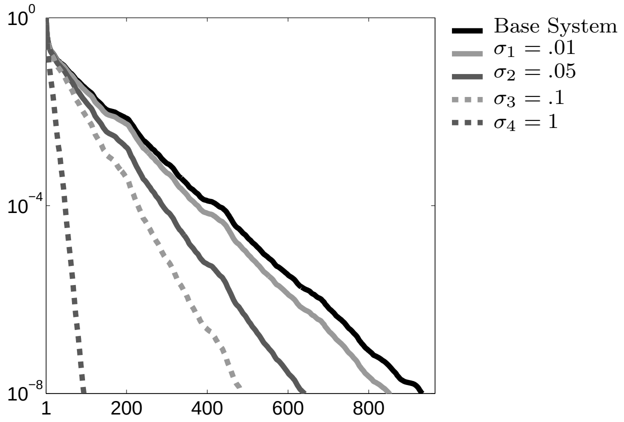

FIG. 2.1. *The simultaneous solution of a base system and four shifted systems using the shifted GMRES method. The base system* $\mathbf{A}$ *is constructed from the matrix* `conf5.0-00l4x4-1000` *from Matrix Market [1], which we call* $\mathbf{D}$*. We form* $\mathbf{A} = (1/\kappa_c + .001)\mathbf{I} - \mathbf{D}$ *for a value* κ_c *provided with the matrix. The matrix* $\mathbf{A}$ *is real-positive. GMRES(30) is used to solve the base system, and the shifted GMRES method produces approximations for the four shifted systems. The shifts are listed in the convergence plot.*

$\left[\overline{\mathbf{H}}_m^{(\sigma)} \quad \mathbf{z}_{m+1}\right] = \mathbf{Q}_{m+1}\mathbf{R}_{m+1}$, then this condition can easily be verified by examining the last diagonal entry of $\mathbf{R}_{m+1}$ to detect numerical singularity. In the case that the solution does not exist, we simply perform one more iteration and check this condition again. It is pointed out in [14], though, that for a positive-real matrix $\mathbf{A}$ (field of values being contained in the right half-plane), restarted GMRES for shifted linear systems computes solutions at every iteration for all shifts $\sigma^{(i)} > 0$ and, in addition, we have $\|\mathbf{r}_m\| \leq \left\|\mathbf{r}_m^{(\sigma_i)}\right\|$ for such shifts. The shifts applied in the setting of QCD yield a family of coefficient matrices which are, in theory, real-positive [14]. In Figure 2.1, we demonstrate the convergence of the shifted GMRES method for simultaneously solving a family of five shifted linear systems.

This is an efficient method for solving families of shifted linear systems, but it is based on restarted GMRES, meaning the method inherits the properties of stagnation and unpredictable convergence exhibited by restarted GMRES, see, e.g., [12, 17, 29]. An attractive idea would be to combine Frommer and Glässner's method with a subspace augmentation or deflation technology. Morgan's GMRES-DR [22] is one candidate, and in [8], this method has been extended to simultaneously solve a family of shifted systems. However, it is restricted to the use of approximate invariant subspaces spanned by harmonic Ritz vectors. We cannot select other subspaces using different criteria. As a result, we cannot use the recycled subspace from one family of shifted linear systems to deflate the next one. In a situation in which we are running multiple QCD simulations, we would like to accelerate convergence by deflating between different base matrices. What we seek is a deflation framework in which we can select approximations from an augmented Krylov subspace *and* enforce a residual collinearity condition.

For $\mathbf{A}$ symmetric, this would be achievable. For symmetric positive definite matrices, the deflated conjugate gradient method [24, 28] has been shown to be effective and has the benefit of producing a residual at step m which is collinear with the newest Lanczos vector. In [28], the authors use the positive definiteness of the coefficient matrix $\mathbf{A}$ to construct an $\mathbf{A}$-orthogonal deflation projector, and it is this $\mathbf{A}$-orthogonality property that is used to prove collinearity. In the case that the matrix $\mathbf{A}$ is nonsymmetric, residual collinearity is not guaranteed. When extending this method to shifted, non-Hermitian systems (using the restarted FOM method of Simoncini [30]), we would not have the benefit of inherent collinearity.

Furthermore, in the case of symmetric, indefinite systems, Kilmer and de Sturler [18] extended the recycled MINRES method [41] to solve multiple shifted linear systems (in this case with complex shifts). Their strategy relies on the fact that in the case of symmetric systems, there is no need to restart. Thus, their procedure produces approximations with non-collinear residuals. Such a procedure is not possible in the case of non-symmetric systems, as it would destroy the collinearity needed at restart.

For nonsymmetric matrices, recycling-type techniques have been previously described. The GCRO method [37] allows the user to select the optimal correction over arbitrary subspaces. de Sturler [38] extends this concept by providing a framework for selecting the optimal subspace to retain from one cycle to the next so as to minimize the error produced by discarding information. This algorithm is called GCROT, where OT stands for optimal truncation. A simplified version of the GCROT approach, based on restarted GMRES (called LGMRES) is presented in [3]. Parks et al. in [23] combine the ideas of [22] and [38] and extend them to a sequence of slowly-changing linear systems. They call their method GCRO-DR (Recycled GMRES).

We briefly review this method, as described in [23]. In order to simplify notation, we drop the subscript i and discuss subspace recycling from the point of view of a single base system. Thus, we are simply solving (1.3). Let us assume that we possess a matrix $\widehat{\mathbf{U}} \in \mathbb{C}^{n\times k}$, whose columns span a deflation subspace. We begin by obtaining an orthonormal basis for the subspace spanned by the columns of $\mathbf{A}\widehat{\mathbf{U}} \in \mathbb{C}^{n\times k}$ using the QR-factorization $\mathbf{A}\widehat{\mathbf{U}} = \mathbf{C}\mathbf{S}$ and assigning $\mathbf{U} = \widehat{\mathbf{U}}\mathbf{S}^{-1}$. Now we have

$$\mathbf{U} \in \mathbb{C}^{n\times k} \quad \text{and} \quad \mathbf{C} = \mathbf{A}\mathbf{U} \quad \text{such that} \quad \mathbf{C}^*\mathbf{C} = \mathbf{I}. \tag{2.8}$$

We run $m-k$ steps of the Arnoldi process, but for the projected operator $(\mathbf{I} - \mathbf{C}\mathbf{C}^*)\mathbf{A}$. This will allow us to construct a solution update $\mathbf{x}_m = \mathbf{x}_0 + \mathbf{t}_m$ such that $\mathbf{t}_m \in \mathcal{S}$, where $\mathcal{S} = \mathcal{R}(\mathbf{U}) + \mathcal{K}_{m-k}((\mathbf{I}-\mathbf{C}\mathbf{C}^*)\mathbf{A}, \mathbf{r}_0)$. For this to work, we require that we begin with an approximation $\mathbf{x}_0$ such that our initial residual $\mathbf{r}_0$ is orthogonal to $\mathcal{R}(\mathbf{C})$. Thus, for an initial approximation $\widetilde{\mathbf{x}}_0$ and initial residual $\widetilde{\mathbf{r}}_0 = \mathbf{b} - \mathbf{A}\widetilde{\mathbf{x}}_0$, we must begin by orthogonally projecting the residual $\widetilde{\mathbf{r}}_0$ onto $\mathcal{R}(\mathbf{C})^\perp$ and then updating $\widetilde{\mathbf{x}}_0$,

$$\mathbf{r}_0 = \widetilde{\mathbf{r}}_0 - \mathbf{C}\mathbf{C}^*\widetilde{\mathbf{r}}_0 \quad \text{and} \quad \mathbf{x}_0 = \widetilde{\mathbf{x}}_0 + \mathbf{U}\mathbf{C}^*\widetilde{\mathbf{r}}_0. \tag{2.9}$$

Let us now generate the Krylov subspace $\mathcal{K}_{m-k}((\mathbf{I}-\mathbf{C}\mathbf{C}^*)\mathbf{A}, \mathbf{r}_0)$ of dimension $m-k$. Let $\mathbf{V}_{m-k} \in \mathbb{C}^{n\times(m-k)}$ have columns which form an orthonormal basis of $\mathcal{K}_{m-k}((\mathbf{I}-\mathbf{C}\mathbf{C}^*)\mathbf{A}, \mathbf{r}_0)$, where we have the usual Arnoldi relation

$$\begin{aligned}(\mathbf{I}-\mathbf{C}\mathbf{C}^*)\mathbf{A}\mathbf{V}_{m-k} &= \mathbf{V}_{m-k+1}\overline{\mathbf{H}}_{m-k}, \qquad \text{i.e.,}\\ \mathbf{A}\mathbf{V}_{m-k} &= \mathbf{C}\mathbf{C}^*\mathbf{A}\mathbf{V}_{m-k} + \mathbf{V}_{m-k+1}\overline{\mathbf{H}}_{m-k}\end{aligned} \tag{2.10}$$

with $\overline{\mathbf{H}}_{m-k} \in \mathbb{C}^{(m-k+1)\times(m-k)}$ being the usual upper Hessenberg matrix. We can rewrite (2.8) together with (2.10) as the Arnoldi-like relation,

$$\mathbf{A}\begin{bmatrix}\mathbf{U} & \mathbf{V}_{m-k}\end{bmatrix} = \begin{bmatrix}\mathbf{C} & \mathbf{V}_{m-k+1}\end{bmatrix}\begin{bmatrix}\mathbf{I} & \mathbf{B}\\ \mathbf{0} & \overline{\mathbf{H}}_{m-k}\end{bmatrix}, \tag{2.11}$$

where $\mathbf{B} = \mathbf{C}^*\mathbf{A}\mathbf{V}_{m-k}$. If we put

$$\widehat{\mathbf{V}}_m = \begin{bmatrix} \mathbf{U} & \mathbf{V}_{m-k} \end{bmatrix}, \; \widehat{\mathbf{W}}_{m+1} = \begin{bmatrix} \mathbf{C} & \mathbf{V}_{m-k+1} \end{bmatrix}, \text{ and } \overline{\mathbf{G}}_m = \begin{bmatrix} \mathbf{I}_k & \mathbf{B} \\ \mathbf{0} & \overline{\mathbf{H}}_{m-k} \end{bmatrix}, \tag{2.12}$$

then we can rewrite (2.11) as

$$\mathbf{A}\widehat{\mathbf{V}}_m = \widehat{\mathbf{W}}_{m+1}\overline{\mathbf{G}}_m. \tag{2.13}$$

At the end of the cycle, we can solve

$$\mathbf{x} = \operatorname*{argmin}_{\mathbf{x}\in\mathbf{x}_0+\mathcal{R}(\widehat{\mathbf{V}}_m)} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|$$

by solving the small $(m+1)\times m$ least-squares problem

$$\mathbf{y}_m = \operatorname*{argmin}_{\mathbf{y}\in\mathbb{C}^m} \left\| \|\mathbf{r}_0\| \, \mathbf{e}_{k+1}^{(m+1)} - \overline{\mathbf{G}}_m\mathbf{y} \right\|, \tag{2.14}$$

and setting

$$\mathbf{x} = \mathbf{x}_0 + \widehat{\mathbf{V}}_m\mathbf{y}_m \;\text{ and }\; \mathbf{r} = \mathbf{r}_0 - \widehat{\mathbf{W}}_{m+1}\overline{\mathbf{G}}_m\mathbf{y}_m. \tag{2.15}$$

However, we do not actually need to solve this least squares problem. We can solve a smaller problem. We can decouple (2.14), and instead solve a small GMRES-like minimization problem. Let

$$\mathbf{y}_m = \begin{bmatrix} \mathbf{y}_{k\times 1} \\ \mathbf{y}_{(m-k)\times 1} \end{bmatrix} \tag{2.16}$$

where $\mathbf{y}_{k\times 1} \in \mathbb{C}^k$ and $\mathbf{y}_{(m-k)\times 1} \in \mathbb{C}^{m-k}$. We can then solve

$$\mathbf{y}_{(m-k)\times 1} = \operatorname*{argmin}_{\mathbf{y}\in\mathbb{C}^{m-k}} \left\| \|\mathbf{r}_0\| \, \mathbf{e}_1^{(m-k+1)} - \overline{\mathbf{H}}_{m-k}\mathbf{y} \right\|,$$

and compute $\mathbf{y}_{k\times 1} = -\mathbf{B}\mathbf{y}_{(m-k)\times 1}$. This is a direct consequence of the upper triangular structure of $\overline{\mathbf{G}}_m$, and was already used in the proof of [37, Theorem 2.2]. Furthermore, by this same theorem, the minimum residual norm

$$\left\| \|\mathbf{r}_0\| \, \mathbf{e}_1^{(m-k+1)} - \overline{\mathbf{H}}_{m-k}\mathbf{y}_{(m-k)\times 1} \right\|$$

is also the residual norm for the approximation computed over the augmented space, thus yielding a cheap residual norm computation. This last fact can also be seen as a consequence of the requirement that $\mathbf{r}_0 \perp \mathcal{R}(\mathbf{C})$. Since we require that

$$\mathbf{r}_m \perp \mathcal{R}(\mathbf{C}) \oplus \mathcal{K}_{m-k+1}((\mathbf{I} - \mathbf{C}\mathbf{C}^*)\mathbf{A}, \mathbf{r}_0),$$

and $\mathbf{r}_0$ is already orthogonal to $\mathcal{R}(\mathbf{C})$, the Recycled GMRES residual update reduces to an update from the projected Krylov subspace. In other words, in the residual update, there is no component from $\mathcal{R}(\mathbf{C})$, and we can rewrite the update (2.15)

$$\mathbf{x} = \mathbf{x}_0 + \mathbf{U}\mathbf{y}_{k\times 1} + \mathbf{V}_{m-k}\mathbf{y}_{(m-k)\times 1} \;\text{ and }\; \mathbf{r} = \mathbf{r}_0 - \mathbf{V}_{m-k+1}\overline{\mathbf{H}}_{m-k}\mathbf{y}_{(m-k)\times 1}. \tag{2.17}$$

At the end of the cycle, we construct a new recycled subspace; and if we have not already converged, we begin the next cycle. If we have converged, we save this

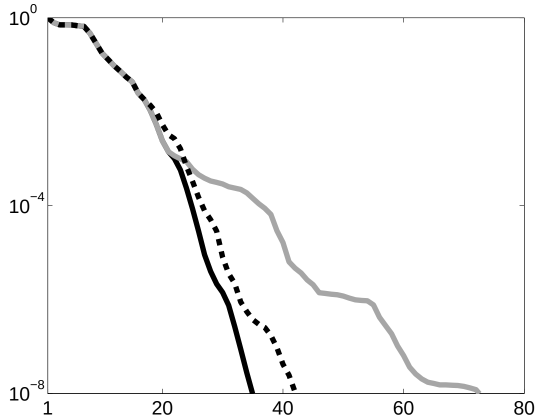

FIG. 2.2. *The convergence curves of GMRES* (***solid black curve***)*, GMRES(20)* (***solid grey curve***)*, and GCRODR(*16,2*)* (***dashed black curve***)*. Observe that the latter two methods have the same storage requirements. The coefficient matrix and right-hand side are from the* `sherman5` *system from Matrix Market [1]. We preconditioned with* `ILU(0)`*. We see that for this problem, it suffices to retain a two-dimensional subspace between restart cycles to achieve convergence almost as good as full GMRES.*

constructed deflation subspace to use when solving the next linear system. In Figure 2.2, we demonstrate the acceleration of convergence of Recycled GMRES with a two-dimensional recycled subspace, as compared to GMRES and GMRES(20). For the recycled subspace, we computed the two harmonic Ritz vectors associated to the harmonic Ritz values of smallest magnitude.

Convergence results for augmented Krylov subspace methods have been previously presented, see, e.g., [10, 25], but not much work has been done in the context of Recycled GMRES. de Sturler has presented some not-yet-published work that specifically addresses the convergence behavior of optimal methods in which we recycle using the above framework [39]. This work asserts that the improvement of convergence bounds from recycling a particular subspace can be quantified according to the quality of the recycled subspace as an invariant subspace. A particular finding, backed up by empirical observation, is that an approximate invariant subspace of modest quality (as judged by the largest principle angle between $\mathbf{U}$ and $\mathbf{C}$) will still yield improvements in bounds on the residual norm.

It should be noted that for a single system, if we have no initial recycled space and compute harmonic Ritz vectors at each restart, GMRES with recycling is algebraically equivalent to Morgan's GMRES-DR [22]. Iterating orthogonally to an approximate invariant subspace to accelerate convergence of GMRES can be justified by the theoretical work in [31]. It was shown that the widely observed two-stage convergence behavior of GMRES, which has been termed *superlinear convergence*, is governed by how well the Krylov subspace approximates a certain eigenspace. Specifically, when the Krylov subspace contains a good approximation to the eigenspace (call this eigenspace $\mathcal{S}$) associated to eigenvalues hindering convergence, we will switch from the slow phase to the fast phase, and convergence will mimic that of GMRES on the projected operator $\mathcal{P}_{\mathcal{S}}^{\perp}\mathbf{A}$ where $\mathcal{P}_{\mathcal{S}}^{\perp}$ is the orthogonal projector onto the orthogonal complement of $\mathcal{S}$. This analysis complements previous discussions of this phenomenon, see e.g., [4, 40].

3. Recycled GMRES For Shifted Systems. Subspace recycling has shown great potential to improve the convergence of restarted methods, in many cases, without dramatically increasing the memory costs. Therefore, if we can incorporate GMRES for shifted linear systems into the recycling framework described in [23], we will have a storage-efficient method which will solve all shifted systems simultaneously but is less likely to suffer from problems which plague restarted GMRES. We denote this method Recycled GMRES *for shifted systems*.

In terms of extending the work of Frommer and Glässner [14], we present an ideal method, in which we construct multiple deflation spaces, one for each shift. However, such a method is memory-intensive. Furthermore, constructing the spaces is computationally expensive. It may be useful when there are only a small number of shifts and when our deflation space is not too large. Unfortunately, without multiple deflation spaces, a method which enforces the collinearity condition on shifted systems over an augmented Krylov subspace based on Recycled GMRES cannot exist using this framework, as we will show.

Consider the pair of linear systems (1.3) and (1.4) and suppose, as in Section 2, that we have $\mathbf{U}$ and $\mathbf{C}$, satisfying (2.8). For initial approximations, we choose $\tilde{\mathbf{x}}_0$ and $\tilde{\mathbf{x}}_0^{(\sigma)}$ so that the initial residuals are collinear, i.e., $\tilde{\mathbf{r}}_0 = \tilde{\beta}_0\tilde{\mathbf{r}}_0^{(\sigma)}$. In Recycled GMRES, the initial projection of the residual (2.9) and resulting update of the solution rely on the relation (2.8) between $\mathbf{U}$ and $\mathbf{C}$. This means that even though we can orthogonally project $\mathbf{r}_0^{(\sigma)}$ onto $\mathcal{R}(\mathbf{C})^\perp$, we cannot easily update the approximation associated with the shifted system. In order to project the residual for the shifted system, we would need $\mathbf{U}^{(\sigma)}$ such that

$$\mathbf{C} = \mathbf{A}\mathbf{U} = (\mathbf{A} + \sigma\mathbf{I})\mathbf{U}^{(\sigma)}. \tag{3.1}$$

This would require an additional k vectors of storage for each shift, and $\mathbf{U}^{(\sigma)}$ can only be obtained with significant work. In the Appendix, we show that for a certain choice of $\mathbf{U}$, we can obtain $\mathbf{U}^{(\sigma)}$ without solving (3.1).

Suppose that for a given $\mathbf{C} \in \mathbb{C}^{n\times k}$ with orthonormal columns, we already have matrices $\mathbf{U}$ and $\mathbf{U}^{(\sigma)}$ satisfying the relation (3.1). We make an observation on the relationship between the subspaces $\mathcal{K}_m((\mathbf{I}-\mathbf{C}\mathbf{C}^*)\mathbf{A}, \mathbf{v})$ and $\mathcal{K}_m((\mathbf{I}-\mathbf{C}\mathbf{C}^*)(\mathbf{A}+\sigma\mathbf{I}), \mathbf{v})$, for any vector $\mathbf{v}$ orthogonal to the range of $\mathbf{C}$.

Proposition 3.1. *Let* $\mathbf{C}$ *be a matrix with orthonormal columns. Then* $\mathbf{v}$ *is in the orthogonal complement of the range of* $\mathbf{C}$*, i.e.,* $\mathbf{C}\mathbf{C}^*\mathbf{v} = \mathbf{0}$ *if and only if*

$$\mathcal{K}_m((\mathbf{I} - \mathbf{C}\mathbf{C}^*)\mathbf{A}, \mathbf{v}) = \mathcal{K}_m((\mathbf{I} - \mathbf{C}\mathbf{C}^*)(\mathbf{A} + \sigma\mathbf{I}), \mathbf{v}) \textit{ for all } m \in \mathbb{N} \tag{3.2}$$

Proof. First, suppose $\mathbf{v} \perp \mathcal{R}(\mathbf{C})$. Since $\mathbf{C}\mathbf{C}^*\mathbf{v} = \mathbf{0}$, we have

$$(\mathbf{I} - \mathbf{C}\mathbf{C}^*)(\mathbf{A} + \sigma\mathbf{I})\mathbf{v} = (\mathbf{I} - \mathbf{C}\mathbf{C}^*)\mathbf{A}\mathbf{v} + \sigma(\mathbf{I} - \mathbf{C}\mathbf{C}^*)\mathbf{v} = (\mathbf{I} - \mathbf{C}\mathbf{C}^*)\mathbf{A}\mathbf{v} + \sigma\mathbf{v}.$$

Therefore, when restricted to vectors orthogonal to the range of $\mathbf{C}$, we have that

$$(\mathbf{I} - \mathbf{C}\mathbf{C}^*)(\mathbf{A} + \sigma\mathbf{I}) = (\mathbf{I} - \mathbf{C}\mathbf{C}^*)\mathbf{A} + \sigma\mathbf{I}.$$

Furthermore, since any $\mathbf{u} \in \mathcal{R}((\mathbf{I} - \mathbf{C}\mathbf{C}^*)(\mathbf{A} + \sigma\mathbf{I}))$ is orthogonal to the range of $\mathbf{C}$, we have

$$\left[(\mathbf{I} - \mathbf{C}\mathbf{C}^*)(\mathbf{A} + \sigma\mathbf{I})\right]^j \mathbf{v} = \left[(\mathbf{I} - \mathbf{C}\mathbf{C}^*)\mathbf{A} + \sigma\mathbf{I}\right]^j \mathbf{v}$$

when applied to any $\mathbf{v} \perp \mathcal{R}(\mathbf{C})$. Thus,

$$\mathcal{K}_m((\mathbf{I} - \mathbf{C}\mathbf{C}^*)(\mathbf{A} + \sigma\mathbf{I}), \mathbf{v}) = \mathcal{K}_m((\mathbf{I} - \mathbf{C}\mathbf{C}^*)\mathbf{A} + \sigma\mathbf{I}, \mathbf{v}) = \mathcal{K}_m((\mathbf{I} - \mathbf{C}\mathbf{C}^*)\mathbf{A}, \mathbf{v}),$$

where the last equality follows from the shift invariance property of Krylov subspaces.

Now, suppose (3.2) holds, and let $m = 2$. Since (3.2) holds for any vector in the subspace, it holds for $\mathbf{u} \in \mathcal{K}_2((\mathbf{I}-\mathbf{CC}^*)\mathbf{A},\mathbf{v}) \setminus \mathcal{K}_1((\mathbf{I}-\mathbf{CC}^*)\mathbf{A},\mathbf{v})$, i.e., we have

$$\mathbf{u} = \alpha_1\mathbf{v} + \alpha_2(\mathbf{I}-\mathbf{CC}^*)\mathbf{A}\mathbf{v} = \beta_1\mathbf{v} + \beta_2(\mathbf{I}-\mathbf{CC}^*)\mathbf{A}\mathbf{v} + \beta_2\sigma(\mathbf{I}-\mathbf{CC}^*)\mathbf{v}$$

where α_2 and β_2 are nonzero. This implies

$$(\alpha_2-\beta_2)(\mathbf{I}-\mathbf{CC}^*)\mathbf{A}\mathbf{v} - \beta_2\sigma(\mathbf{I}-\mathbf{CC}^*)\mathbf{v} = (\beta_1-\alpha_1)\mathbf{v},$$

and thus, $\mathbf{v} \perp \mathcal{R}(\mathbf{C})$. □

Thus, suppose we construct an augmented subspace $\mathcal{R}(\mathbf{U})+\mathcal{K}_{m-k}((\mathbf{I}-\mathbf{CC}^*)\mathbf{A},\mathbf{r}_0)$ for the base system and minimize the residual as described in [23]. From the augmented subspace $\mathcal{R}(\mathbf{U}^{(\sigma)}) + \mathcal{K}_{m-k}((\mathbf{I}-\mathbf{CC}^*)\mathbf{A},\mathbf{r}_0)$, we can construct an approximation for the shifted system by enforcing a collinearity condition on the residual of the shifted system. Thus, we can simultaneously solve both the base and shifted systems over augmented Krylov subspaces. Recall that in the presence of no deflation subspace, we have the two Arnoldi relations

$$\mathbf{A}\mathbf{V}_m = \mathbf{V}_{m+1}\overline{\mathbf{H}}_m \quad\text{and}\quad (\mathbf{A}+\sigma\mathbf{I})\mathbf{V}_m = \mathbf{V}_{m+1}\overline{\mathbf{H}}_m^{(\sigma)},$$

respectively for the base and shifted systems, where $\overline{\mathbf{H}}_m^{(\sigma)} = \overline{\mathbf{H}}_m + \begin{bmatrix}\sigma\mathbf{I}_m\\ \mathbf{0}\end{bmatrix}$. From here, it is straightforward to construct an Arnoldi-like relation for the shifted system in the framework of subspace recycling,

$$(\mathbf{A}+\sigma\mathbf{I})\begin{bmatrix}\mathbf{U}^{(\sigma)} & \mathbf{V}_{m-k}\end{bmatrix} = \begin{bmatrix}\mathbf{C} & \mathbf{V}_{m-k+1}\end{bmatrix}\begin{bmatrix}\mathbf{I}_k & \mathbf{B}\\ \mathbf{0} & \overline{\mathbf{H}}_{m-k}^{(\sigma)}\end{bmatrix}. \tag{3.3}$$

We note that the same matrix $\mathbf{B}$ appears in (3.3) as in (2.11) since

$$\begin{aligned}(\mathbf{I}-\mathbf{CC}^*)(\mathbf{A}+\sigma\mathbf{I})\mathbf{V}_{m-k} &= \mathbf{V}_{m-k+1}\overline{\mathbf{H}}_{m-k}^{(\sigma)}\\ (\mathbf{A}+\sigma\mathbf{I})\mathbf{V}_{m-k} - \mathbf{CC}^*\mathbf{A}\mathbf{V}_{m-k} - \sigma\mathbf{CC}^*\mathbf{V}_{m-k} &= \mathbf{V}_{m-k+1}\overline{\mathbf{H}}_{m-k}^{(\sigma)}\\ (\mathbf{A}+\sigma\mathbf{I})\mathbf{V}_{m-k} &= \mathbf{C}\mathbf{B} + \mathbf{V}_{m-k+1}\overline{\mathbf{H}}_{m-k}^{(\sigma)}, \quad (3.4)\end{aligned}$$

where $\mathbf{B} = \mathbf{C}^*\mathbf{A}\mathbf{V}_{m-k}$ and where we used the fact that $\sigma\mathbf{CC}^*\mathbf{V}_{m-k} = \mathbf{0}$. This is due to the fact that the columns of $\mathbf{V}_{m-k}$ are orthogonal to $\mathcal{R}(\mathbf{C})$. We can follow the same arguments as in [14] and construct an approximation for the shifted system such that the residual is collinear to the one produced by Recycled GMRES for the base system,

$$\begin{aligned}\mathbf{r}_m^{(\sigma)} &= \beta_m\mathbf{r}_m \quad (3.5)\\ \mathbf{b} - (\mathbf{A}+\sigma\mathbf{I})\left(\mathbf{x}_0^{(\sigma)} + \begin{bmatrix}\mathbf{U}^{(\sigma)} & \mathbf{V}_{m-k}\end{bmatrix}\mathbf{y}_m^{(\sigma)}\right) &= \beta_m\begin{bmatrix}\mathbf{C} & \mathbf{V}_{m-k+1}\end{bmatrix}\mathbf{z}_m\\ \mathbf{r}_0^{(\sigma)} - \begin{bmatrix}\mathbf{C} & \mathbf{V}_{m-k+1}\end{bmatrix}\begin{bmatrix}\mathbf{I}_k & \mathbf{B}\\ \mathbf{0} & \overline{\mathbf{H}}_{m-k}^{(\sigma)}\end{bmatrix}\mathbf{y}_m^{(\sigma)} &= \beta_m\begin{bmatrix}\mathbf{C} & \mathbf{V}_{m-k+1}\end{bmatrix}\mathbf{z}_m\\ \beta_0\mathbf{r}_0 - \begin{bmatrix}\mathbf{C} & \mathbf{V}_{m-k+1}\end{bmatrix}\begin{bmatrix}\mathbf{I}_k & \mathbf{B}\\ \mathbf{0} & \overline{\mathbf{H}}_{m-k}^{(\sigma)}\end{bmatrix}\mathbf{y}_m^{(\sigma)} &= \beta_m\begin{bmatrix}\mathbf{C} & \mathbf{V}_{m-k+1}\end{bmatrix}\mathbf{z}_m\\ \beta_0\left\|\mathbf{r}_0\right\|\mathbf{e}_{m+1}^{(k+1)} - \begin{bmatrix}\mathbf{I}_k & \mathbf{B}\\ \mathbf{0} & \overline{\mathbf{H}}_{m-k}^{(\sigma)}\end{bmatrix}\mathbf{y}_m^{(\sigma)} &= \beta_m\mathbf{z}_m,\end{aligned}$$

which yields the linear system

$$\begin{bmatrix}\overline{\mathbf{G}}_m^{(\sigma)} & \mathbf{z}_m\end{bmatrix}\begin{bmatrix}\mathbf{y}_m^{(\sigma)}\\ \beta_m\end{bmatrix} = \beta_0 \left\|\mathbf{r}_0\right\| \mathbf{e}_{k+1}^{(m+1)}, \tag{3.6}$$

where

$$\overline{\mathbf{G}}_m^{(\sigma)} = \begin{bmatrix}\mathbf{I}_k & \mathbf{B}\\ \mathbf{0} & \overline{\mathbf{H}}_{m-k}^{(\sigma)}\end{bmatrix}.$$

We then update

$$\mathbf{x}_m^{(\sigma)} = \mathbf{x}_0^{(\sigma)} + \begin{bmatrix}\mathbf{U}^{(\sigma)} & \mathbf{V}_{m-k}\end{bmatrix}\mathbf{y}_m^{(\sigma)}.$$

Observe that the approximation with collinear residual is drawn from a different augmented Krylov subspace than the minimum residual approximation of the base system, i.e., for $\mathbf{U}^{(\sigma)}$ rather than $\mathbf{U}$. As it can be appreciated, this is a straightforward extension of the work of Frommer and Glässner [14].

4. Non-existence of Collinear Residuals.

The algorithm presented in Section 3 is the most natural extension of the recycling framework to GMRES for shifted systems, but in terms of storage, it is not efficient since storage requirements increase for each new shift. Therefore, to develop an algorithm with fixed-storage requirements regardless of the number of shifts, we propose to only store the matrix $\mathbf{U} \in \mathbb{C}^{n\times k}$ whose columns span the recycled subspace associated to the base system. Can we still extend the mechanics of shifted GMRES to this setting? The answer, in general, is no.

The simple appearance of the shift in the Arnoldi relation, as in (2.3), no longer holds in the Arnoldi-like relation (2.13) from GMRES with recycling. Observe that for $\widehat{\mathbf{V}}_m$ and $\widehat{\mathbf{W}}_{m+1}$, defined as in (2.12),

$$(\mathbf{A}+\sigma\mathbf{I})\widehat{\mathbf{V}}_m = \widehat{\mathbf{W}}_{m+1}\begin{bmatrix}\mathbf{I}_k & \mathbf{B}\\ \mathbf{0} & \overline{\mathbf{H}}_{m-k}\end{bmatrix} + \sigma\widehat{\mathbf{V}}_m.$$

If we have

$$\mathrm{Range}(\widehat{\mathbf{V}}_m) \subset \mathrm{Range}(\widehat{\mathbf{W}}_{m+1}), \tag{4.1}$$

the relation could be easily modified so that a relation similar to (2.3) holds, allowing the collinearity condition to be enforced. However, this inclusion, in general, does not hold; the columns of $\mathbf{U}$ span an approximate invariant subspace of $\mathbf{A}$, not a true invariant subspace. In general, the span of the columns of $\mathbf{U}$ will be different than that of $\mathbf{C}$. The same is true of $\widehat{\mathbf{V}}_m$ and $\widehat{\mathbf{W}}_{m+1}$, respectively. Similar observations are made in the context of Hermitian systems in [18].

There is one scenario in which (4.1) does hold. Consider the situation in which we begin with no starting recycled space and compute harmonic Ritz vectors at the end of each cycle to pass into the next cycle. We run an m-step cycle of shifted GMRES, and at the end of that cycle, let the columns of $\mathbf{U}$ be k harmonic Ritz vectors, we compute $\mathbf{C}$ as before, and restart. Morgan [21] showed that for a harmonic Ritz pair $(\mathbf{g}, \theta)$, the eigenvector residual $\mathbf{Ag} - \theta\mathbf{g}$ is a multiple of the GMRES residual $\mathbf{r}_m$. At the end of a cycle, if we compute k harmonic Ritz vectors and store them as the columns of $\widetilde{\mathbf{U}}$, then we know that

$$\mathrm{Range}(\mathbf{A}\widetilde{\mathbf{U}} - \widetilde{\mathbf{U}}\mathbf{D}) = \mathrm{span}(\mathbf{r}_m), \tag{4.2}$$

where $\mathbf{D} = \operatorname{diag}(\theta_1, \ldots, \theta_k)$, the diagonal matrix containing the respective harmonic Ritz values associated to the columns of $\widetilde{\mathbf{U}}$. If we compute the QR-factorization of $\mathbf{A}\widetilde{\mathbf{U}} = \mathbf{CR}$ and let $\mathbf{U} = \widetilde{\mathbf{U}}\mathbf{R}^{-1}$, then for $\mathbf{T} = \mathbf{RDR}^{-1}$ we have

$$\operatorname{Range}(\mathbf{C} - \mathbf{UT}) = \operatorname{span}(\mathbf{r}_m).$$

At the beginning of the next cycle, we take

$$\mathbf{v}_1 = \mathbf{r}_m / \left\|\mathbf{r}_m\right\| \tag{4.3}$$

as the first Krylov vector; and in this case, the containment (4.1) holds. This is the same fact exploited in [8], where the authors observe that the augmented Krylov subspace is itself actually a larger Krylov subspace with a different starting vector. Thus, the shifted GMRES method can be applied directly to the Krylov subspace augmented with the harmonic Ritz vectors, as long as there was no deflation space at the beginning of the process.

What about in the general setting? Let $\mathbf{E}$ be a matrix whose columns form a basis for the orthogonal complement of $\mathcal{R}(\mathbf{C}) \oplus \mathcal{R}(\mathbf{V}_{m-k+1})$ in $\mathbb{R}^n$. We note that $\mathbf{E}$ needs not be computed; we use it here as a theoretical tool to develop our algorithm. We can write

$$\mathbf{U} = \mathbf{CY} + \mathbf{V}_{m-k+1}\mathbf{Z} + \mathbf{EF}, \tag{4.4}$$

where $\mathbf{Y} \in \mathbb{C}^{k\times k}$, $\mathbf{Z} \in \mathbb{C}^{(m-k+1)\times k}$, and $\mathbf{F} \in \mathbb{C}^{(n-m-1)\times k}$. This yields the following *imperfect* Arnoldi-like relation for the shifted system,

$$(\mathbf{A} + \sigma\mathbf{I})\begin{bmatrix}\mathbf{U} & \mathbf{V}_{m-k}\end{bmatrix} = \begin{bmatrix}\mathbf{C} & \mathbf{V}_{m-k+1}\end{bmatrix}\begin{bmatrix}\mathbf{I}_k + \sigma\mathbf{Y} & \mathbf{B} \\ \sigma\mathbf{Z} & \overline{\mathbf{H}}_{m-k}^{(\sigma)}\end{bmatrix} + \sigma\begin{bmatrix}\mathbf{EF} & \mathbf{0}\end{bmatrix}. \tag{4.5}$$

If we let

$$\widetilde{\mathbf{G}}_m^{(\sigma)} = \begin{bmatrix}\mathbf{I}_k + \sigma\mathbf{Y} & \mathbf{B} \\ \sigma\mathbf{Z} & \overline{\mathbf{H}}_{m-k}^{(\sigma)}\end{bmatrix}, \tag{4.6}$$

together with (2.13), then the Arnoldi-like relation (4.5) can be rewritten as

$$(\mathbf{A} + \sigma\mathbf{I})\widehat{\mathbf{V}}_m = \widehat{\mathbf{W}}_{m+1}\widetilde{\mathbf{G}}_m^{(\sigma)} + \sigma\begin{bmatrix}\mathbf{EF} & \mathbf{0}\end{bmatrix}.$$

At the end of the cycle, we minimize the residual of the base system accordingly, obtaining $\mathbf{y}_m$ as in (2.16), so that

$$\begin{aligned}\mathbf{r}_m &= \mathbf{r}_0 - \widehat{\mathbf{W}}_{m+1}\overline{\mathbf{G}}_m\mathbf{y}_m \\ &= \left\|\mathbf{r}_0\right\|\widehat{\mathbf{W}}_{m+1}\mathbf{e}_{m+1}^{(k+1)} - \widehat{\mathbf{W}}_{m+1}\overline{\mathbf{G}}_m\mathbf{y}_m \\ &= \widehat{\mathbf{W}}_{m+1}\mathbf{z}_{m+1},\end{aligned}$$

where

$$\mathbf{z}_{m+1} = \left\|\mathbf{r}_0\right\|\mathbf{e}_{k+1}^{(m+1)} - \overline{\mathbf{G}}_m\mathbf{y}_m. \tag{4.7}$$

Now, for the shifted system, we would like to enforce the collinearity condition. If a collinear residual were to exist for the shifted system, then it would satisfy

$$\begin{aligned}
\mathbf{r}_m^{(\sigma)} &= \beta_m \mathbf{r}_m \\
\mathbf{b} - (\mathbf{A}+\sigma\mathbf{I})(\mathbf{x}_0^{(\sigma)} + \widehat{\mathbf{V}}_m \mathbf{y}_m^{(\sigma)}) &= \beta_m \widehat{\mathbf{W}}_{m+1}\mathbf{z}_{m+1} \\
\mathbf{r}_0^{(\sigma)} - (\mathbf{A}+\sigma\mathbf{I})\widehat{\mathbf{V}}_m \mathbf{y}_m^{(\sigma)} &= \widehat{\mathbf{W}}_{m+1}\mathbf{z}_{m+1}\beta_m \\
\beta_0\mathbf{r}_0 - (\widehat{\mathbf{W}}_{m+1}\widetilde{\mathbf{G}}_m^{(\sigma)} + \sigma\begin{bmatrix}\mathbf{EF} & \mathbf{0}\end{bmatrix})\mathbf{y}_m^{(\sigma)} &= \widehat{\mathbf{W}}_{m+1}\mathbf{z}_{m+1}\beta_m \\
\beta_0\mathbf{r}_0 &= \widehat{\mathbf{W}}_{m+1}(\mathbf{z}_{m+1}\beta_m + \widetilde{\mathbf{G}}_m^{(\sigma)}\mathbf{y}_m^{(\sigma)}) \\
&\quad +\sigma\begin{bmatrix}\mathbf{EF} & \mathbf{0}\end{bmatrix}\mathbf{y}_m^{(\sigma)}. \qquad (4.8)
\end{aligned}$$

However, the shifted approximation yielding collinear residual exists only when $\sigma\begin{bmatrix}\mathbf{EF} & \mathbf{0}\end{bmatrix} = \mathbf{0}$. This is not generally satisfied when (4.1) is false. Observe that in the general case, $\mathbf{r}_0 \in \mathcal{R}(\mathbf{C}) \oplus \mathcal{R}(\mathbf{V}_{m-k+1})$ while the right-hand side of (4.8) has a non-zero component in $\mathcal{R}(\mathbf{E}) = (\mathcal{R}(\mathbf{C}) \oplus \mathcal{R}(\mathbf{V}_{m-k+1}))^{\perp}$. Thus, we state the conditions for existence (and nonexistence) of the collinear residual in the following theorem.

Theorem 4.1. *Let $\mathbf{U}$ and $\mathbf{C}$ be defined as in* (2.8)*. Suppose we have approximations $\mathbf{x}_0$ and $\mathbf{x}_0^{(\sigma)}$ to the solutions of* (1.3) *and* (1.4)*, respectively, such that the residuals $\mathbf{r}_0$ and $\mathbf{r}_0^{(\sigma)}$ are collinear, i.e., $\mathbf{r}_0^{(\sigma)} = \beta_0\mathbf{r}_0$, and $\mathbf{r}_0 \perp \mathcal{R}(\mathbf{C})$. Let $\mathbf{r}_m$ be the minimum residual solution produced by Recycled GMRES over the augmented Krylov subspace $\mathcal{R}(\mathbf{U}) + \mathcal{K}_{m-k}((\mathbf{I}-\mathbf{CC}^*)\mathbf{A}, \mathbf{r}_0)$. Then one of the following is true:*

- $\mathcal{R}(\mathbf{U}) + \mathcal{K}_{m-k}((\mathbf{I}-\mathbf{CC}^*)\mathbf{A}, \mathbf{r}_0) \subset \mathcal{R}(\mathbf{C}) \oplus \mathcal{K}_{m-k+1}((\mathbf{I}-\mathbf{CC}^*)\mathbf{A}, \mathbf{r}_0)$
- *There exists **no** approximation $\mathbf{x}_m^{(\sigma)} \in \mathcal{R}(\mathbf{U}) + \mathcal{K}_{m-k}((\mathbf{I}-\mathbf{CC}^*)\mathbf{A}, \mathbf{r}_0)$ to* (1.4) *such that $\mathbf{r}_m^{(\sigma)}$ is collinear to $\mathbf{r}_m$, i.e., $\mathbf{r}_m^{(\sigma)} \neq \beta_m\mathbf{r}_m$, for **all** $\beta_m \in \mathbb{C}$.*

5. An Approximate Collinearity Scheme. In general, Theorem 4.1 states we do not have collinear residuals. What is the best we can do? Before proceeding with the mathematical details, it will be helpful to provide an overview of the strategy we are proposing and encode this into a schematic algorithm. This algorithm will allow us to get a good approximation for the shifted system at low cost. As before, the base system will be solved using cycles of GMRES with recycling. For the shifted system, though we can initially project the residual away from $\mathcal{R}(\mathbf{C})$, we cannot perform the corresponding update of the solution. Instead, at Line 5 of Algorithm 5.1, we perform an update of the shifted system approximation which implicitly updates the residual by the perturbation of an orthogonal projection. We have

$$\mathbf{x}_0^{(\sigma)} = \widetilde{\mathbf{x}}_0^{(\sigma)} + \mathbf{UC}^*\widetilde{\mathbf{r}}_0^{(\sigma)}.$$

The updated residual can be written as

$$\begin{aligned}
\mathbf{r}_0^{(\sigma)} &= \mathbf{b} - (\mathbf{A}+\sigma\mathbf{I})\mathbf{x}_0^{(\sigma)} \\
&= \mathbf{b} - (\mathbf{A}+\sigma\mathbf{I})(\widetilde{\mathbf{x}}_0^{(\sigma)} + \mathbf{UC}^*\widetilde{\mathbf{r}}_0^{(\sigma)}) \\
&= \widetilde{\mathbf{r}}_0^{(\sigma)} - (\mathbf{A}+\sigma\mathbf{I})\mathbf{UC}^*\widetilde{\mathbf{r}}_0^{(\sigma)} \\
&= \underbrace{\widetilde{\mathbf{r}}_0^{(\sigma)} - \mathbf{CC}^*\widetilde{\mathbf{r}}_0^{(\sigma)}}_{\text{true orthogonal projection}} - \underbrace{\sigma\mathbf{UC}^*\widetilde{\mathbf{r}}_0^{(\sigma)}}_{\textit{perturbation}} \qquad (5.1)
\end{aligned}$$

We also cannot enforce a residual collinearity condition at the end of each cycle. However, neglecting a term from the collinearity equation allows us to solve a nearby approximate collinearity condition directly. We use the solution from this nearby equation to update the approximation for the shifted system. These corrections to the shifted system solution improve the residual but do not lead to convergence for the shifted system which will start with an expected improved approximation. We present analysis showing how much improvement is possible with this method. We terminate the iteration when the base system has converged. The algorithm can be applied recursively on the remaining unconverged shifted systems. This recursive method of solving one seed system at a time while choosing corrections for the approximations for the other systems has been previously suggested in the context of linear systems with multiple right-hand sides, see e.g., [6, 34].

We present an outline of this process in Algorithm 5.1. Note, in the presentation of the algorithm, we solve multiple shifted systems in order to demonstrate the algorithm's amenability to recursion. We describe Algorithm 5.1 in more detail.

As we previously stated, **EF** is generally non-zero, meaning the collinear residual

Algorithm 5.1: Schematic of Shifted Recycled GMRES with an Approximate Collinearity Condition

```
Input : $\mathbf{A} \in \mathbb{C}^{n\times n}$; $\left\{\sigma^{(\ell)}\right\}_{\ell=1}^{L} \subset \mathbb{C}$; $\mathbf{U}, \mathbf{C} \in \mathbb{C}^{n\times k}$ such that $\mathbf{AU} = \mathbf{C}$ and
        $\mathbf{C}^*\mathbf{C} = \mathbf{I}_k$; Initial Approximations $\mathbf{x}_0$ and $\mathbf{x}_0^{(\sigma^{(\ell)})}$ such that residuals
        are collinear; $\varepsilon > 0$
$\mathbf{x} \leftarrow \mathbf{x}_0$, $\mathbf{r} = \mathbf{b} - \mathbf{Ax}$
$\mathbf{x} \leftarrow \mathbf{x} + \mathbf{UC}^*\mathbf{r}$, $\mathbf{r} \leftarrow \mathbf{r} - \mathbf{CC}^*\mathbf{r}$;          Project base residual
$\mathbf{x}^{(\sigma^{(\ell)})} \leftarrow \mathbf{x}_0^{(\sigma^{(\ell)})}$, $r^{(\sigma^{(\ell)})} = \mathbf{b} - \mathbf{Ax}^{(\sigma^{(\ell)})}$ for all $\ell$
for $\ell = 1$ to $L$ do
    $\mathbf{x}^{(\sigma^{(\ell)})} \leftarrow \mathbf{x}^{(\sigma^{(\ell)})} + \mathbf{UC}^*\mathbf{r}^{(\sigma^{(\ell)})}$; Update shifted approximation, but not
    an implicit residual projection
while $\|\mathbf{r}\| > \varepsilon$ do
    Construct a basis of the subspace $\mathcal{K}_m((\mathbf{I} - \mathbf{CC}^*)\mathbf{A}, \mathbf{r})$
    Compute update $\mathbf{t} \in \mathrm{Range}(\mathbf{U}) + \mathcal{K}_m((\mathbf{I} - \mathbf{CC}^*)\mathbf{A}, \mathbf{r})$ by minimizing
    residual using Recycled GMRES
    $\mathbf{x} \leftarrow \mathbf{x} + \mathbf{t}$; $\mathbf{r} \leftarrow \mathbf{b} - \mathbf{Ax}$
    for $\ell = 1$ to $L$ do
        Compute update $\mathbf{t}^{(\sigma^{(\ell)})} \in \mathrm{Range}(\mathbf{U}) + \mathcal{K}_m((\mathbf{I} - \mathbf{CC}^*)\mathbf{A}, \mathbf{r})$ according
        to the approximate collinearity condition
        $\mathbf{x}_\ell^{(\sigma)} \leftarrow \mathbf{x}_\ell^{(\sigma)} + \mathbf{t}_\ell^{(\sigma)}$
    Compute updated recycled subspace information $\mathbf{U}$ and $\mathbf{C}$
Clear any variables no longer needed
if $L > 2$ then
    Make a recursive call to Algorithm 5.1 with $\mathbf{A} \leftarrow \mathbf{A} + \sigma^{(1)}\mathbf{I}$, shifts
    $\left\{\sigma^{(\ell)} - \sigma^{(1)}\right\}_{\ell=2}^{L}$, approximations $\left\{\mathbf{x}^{\sigma^{(\ell)}}\right\}_{\ell=2}^{L}$ and updated recycled
    subspace matrix $\mathbf{U}$
else
    Apply Recycled GMRES to $\mathbf{x}^{(\sigma^{(L)})}$ yielding shifted system approximation
    that achieves tolerance $\varepsilon$
```

does not exist. However, if it is relatively small (e.g., in the sense of the 2-norm) perhaps we can disregard it. This yields an augmented linear system which can be solved directly,

$$\mathbf{z}_{m+1}\tilde{\beta}_m + \widetilde{\mathbf{G}}_m^{(\sigma)}\tilde{\mathbf{y}}_m^{(\sigma)} = \beta_0 \left\|\mathbf{r}\right\| \mathbf{e}_{k+1}^{(m+1)}, \quad \text{or} \tag{5.2}$$
$$\begin{bmatrix} \widetilde{\mathbf{G}}_m^{(\sigma)} & \mathbf{z}_{m+1} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{y}}_m^{(\sigma)} \\ \tilde{\beta}_m \end{bmatrix} = \beta_0 \left\|\mathbf{r}\right\| \mathbf{e}_{k+1}^{(m+1)}.$$

Thus, we proceed by solving this nearby problem and updating the shifted solution,

$$\mathbf{x}_m^{(\sigma)} = \mathbf{x}_0^{(\sigma)} + \widehat{\mathbf{V}}_m\tilde{\mathbf{y}}_m^{(\sigma)}. \tag{5.3}$$

For each restart cycle, we repeat this process for the shifted system. We stop when the base residual norm is below tolerance. As experiments show in Section 6, the update (5.3) at each cycle improves the solution for the shifted system, at very little additional cost. When the residual norm for the base system reaches tolerance, the residual norm of the shifted system will have been reduced but, generally, not enough to satisfy the given tolerance. We then apply the GMRES with recycling algorithm with this approximate collinearity scheme to the remaining unsolved systems, taking one of the shifted systems as our new base system. Thus, this method is amenable to recursion on the number of shifts. If we have only one system remaining to solve, we simply use Recycled GMRES.

Observe that for any number of shifts, we can easily form $\widetilde{\mathbf{G}}_m^{(\sigma)}$ for each σ at little additional cost. The matrices $\mathbf{Y}$ and $\mathbf{Z}$ in (4.4) must be computed only once per iteration, regardless of the number of shifted systems we are solving. However, additional shifts will require more recursive calls to the algorithm and, thus, more iterations.

5.1. Analysis of the Approximate Collinearity Condition. We provide a simple analysis which justifies why the approximate collinearity condition can produce improved approximations to the solutions of the shifted systems and to understand how well we can expect the algorithm to perform. This analysis also yields a cheap way in which we can monitor the progress of the residuals of the shifted systems.

THEOREM 5.1. *Suppose we begin the cycle as in* (4.8)*, with approximate collinearity between the base and shifted residuals, satisfying the relation*

$$\mathbf{r}_0^{(\sigma)} = \tilde{\beta}_0\mathbf{r}_0 + \mathbf{s}^{(\sigma)}. \tag{5.4}$$

Then if we perform a cycle of Recycled GMRES to reduce the residual of the base system and apply the approximate collinearity condition (5.2)*, we have the relation*

$$\tilde{\mathbf{r}}_m^{(\sigma)} = \tilde{\beta}_m\mathbf{r}_m - \sigma\mathbf{E}\mathbf{F}\left(\tilde{\mathbf{y}}_m^{(\sigma)}\right)_{1:k} + \mathbf{s}^{(\sigma)}. \tag{5.5}$$

Proof. We can write the residual produced by the approximate collinearity procedure for the shifted system,

$$\begin{aligned}
\tilde{\mathbf{r}}_m^{(\sigma)} &= \mathbf{b} - (\mathbf{A} + \sigma\mathbf{I})\mathbf{x}_m^{(\sigma)} \qquad (5.6)\\
&= \mathbf{b} - (\mathbf{A} + \sigma\mathbf{I})(\mathbf{x}_0^{(\sigma)} + \widehat{\mathbf{V}}_m \widetilde{\mathbf{y}}_m^{(\sigma)})\\
&= \mathbf{r}_0^{(\sigma)} - (\mathbf{A} + \sigma\mathbf{I})\widehat{\mathbf{V}}_m \widetilde{\mathbf{y}}_m^{(\sigma)}\\
&= \tilde{\beta}_0 \mathbf{r}_0 + \mathbf{s}^{(\sigma)} - (\mathbf{A} + \sigma\mathbf{I})\widehat{\mathbf{V}}_m \tilde{\mathbf{y}}_m^{(\sigma)}\\
&= \tilde{\beta}_0 \mathbf{r}_0 - \left(\widehat{\mathbf{W}}_{m+1}\widetilde{\mathbf{G}}_m^{(\sigma)} + \sigma \begin{bmatrix}\mathbf{EF} & \mathbf{0}\end{bmatrix}\right) \tilde{\mathbf{y}}_m^{(\sigma)} + \mathbf{s}^{(\sigma)}\\
&= \tilde{\beta}_0 \left\|\mathbf{r}_0\right\| \widehat{\mathbf{W}}_{m+1}\mathbf{e}_{m+1}^{(k+1)} - \widehat{\mathbf{W}}_{m+1}\widetilde{\mathbf{G}}_m^{(\sigma)}\tilde{\mathbf{y}}_m^{(\sigma)} - \sigma \begin{bmatrix}\mathbf{EF} & \mathbf{0}\end{bmatrix} \tilde{\mathbf{y}}_m^{(\sigma)} + \mathbf{s}^{(\sigma)}\\
&= \tilde{\beta}_0 \left\|\mathbf{r}_0\right\| \widehat{\mathbf{W}}_{m+1}\mathbf{e}_{m+1}^{(k+1)} - \widehat{\mathbf{W}}_{m+1}\widetilde{\mathbf{G}}_m^{(\sigma)}\tilde{\mathbf{y}}_m^{(\sigma)} - \tilde{\beta}_m \widehat{\mathbf{W}}_{m+1}\mathbf{z}_{m+1} + \tilde{\beta}_m \widehat{\mathbf{W}}_{m+1}\mathbf{z}_{m+1}\\
&\quad -\sigma \begin{bmatrix}\mathbf{EF} & \mathbf{0}\end{bmatrix} \tilde{\mathbf{y}}_m^{(\sigma)} + \mathbf{s}^{(\sigma)}\\
&= \widehat{\mathbf{W}}_{m+1}\left(\tilde{\beta}_0 \left\|\mathbf{r}_0\right\| \mathbf{e}_{m+1}^{(k+1)} - \widetilde{\mathbf{G}}_m^{(\sigma)}\tilde{\mathbf{y}}_m^{(\sigma)} - \tilde{\beta}_m \mathbf{z}_{m+1}\right) + \tilde{\beta}_m \widehat{\mathbf{W}}_{m+1}\mathbf{z}_{m+1}\\
&\quad -\sigma \begin{bmatrix}\mathbf{EF} & \mathbf{0}\end{bmatrix} \tilde{\mathbf{y}}_m^{(\sigma)} + \mathbf{s}^{(\sigma)}.
\end{aligned}$$

Now using the approximate collinearity condition (5.2) and the fact that by definition $\mathbf{r}_m = \widehat{\mathbf{W}}_{m+1}\mathbf{z}_{m+1}$, we have that

$$\tilde{\mathbf{r}}_m^{(\sigma)} = \tilde{\beta}_m \mathbf{r}_m - \sigma \begin{bmatrix}\mathbf{EF} & \mathbf{0}\end{bmatrix} \tilde{\mathbf{y}}_m^{(\sigma)} + \mathbf{s}^{(\sigma)}. \qquad (5.7)$$

Then we can write

$$\tilde{\mathbf{r}}_m^{(\sigma)} = \tilde{\beta}_m \mathbf{r}_m - \sigma \mathbf{EF} \left(\tilde{\mathbf{y}}_m^{(\sigma)}\right)_{1:k} + \mathbf{s}^{(\sigma)}. \qquad \square$$

It should be noted that the term $-\sigma \mathbf{EF} \left(\tilde{\mathbf{y}}_m^{(\sigma)}\right)_{1:k} + \mathbf{s}^{(\sigma)}$ is a function of the quality of the recycled subspaces as well as of σ. With the use of simple inequalities, we obtain an important corollary describing the amount of residual norm reduction we can expect for the shifted systems.

COROLLARY 5.2. *Under the same assumptions as in Theorem* 5.1*, if we apply the approximate collinearity condition* (5.2) *to obtain a correction* $\mathbf{x}_m^{(\sigma)}$ *for the shifted system, then the associated residual satisfies the following inequality,*

$$\left\|\tilde{\mathbf{r}}_m^{(\sigma)}\right\| \leq \left|\tilde{\beta}_m\right| \left\|\mathbf{r}_m\right\| + |\sigma| \left\|\mathbf{EF}\right\| \left\|\left(\tilde{\mathbf{y}}_m^{(\sigma)}\right)_{1:k}\right\| + \left\|\mathbf{s}^{(\sigma)}\right\|.$$

This gives us important information about the performance of this method with regard to the shifted system. The norm of $\tilde{\mathbf{r}}_m^{(\sigma)}$ is dominated by a scaled multiple of the norm of $\mathbf{r}_m$ as long as $\left|\tilde{\beta}_m\right| \left\|\mathbf{r}_m\right\|$ is larger compared to $|\sigma| \left\|\mathbf{EF}\right\| \left\|\left(\tilde{\mathbf{y}}_m^{(\sigma)}\right)_{1:k}\right\| + \left\|\mathbf{s}^{(\sigma)}\right\|$. As long as this is the case, we can expect a decrease in the norm of $\mathbf{r}_m$ to yield a decrease in the norm of $\tilde{\mathbf{r}}_m^{(\sigma)}$ Furthermore, we see that the amount of improvement we can expect from computing corrections for the shifted system according to (5.2) is affected by $|\sigma|$, $\left\|\mathbf{EF}\right\|$, and $\left\|\left(\tilde{\mathbf{y}}_m^{(\sigma)}\right)_{1:k}\right\|$. We cannot control $\left\|\left(\tilde{\mathbf{y}}_m^{(\sigma)}\right)_{1:k}\right\|$, and σ is dictated by the problem. However, the size of $\left\|\mathbf{EF}\right\|$ is connected to the quality of $\mathbf{U}$ as an approximation to an invariant subspace of $\mathbf{A}$. This can seen by writing

$$\mathbf{EF} = \mathbf{U} - (\mathbf{CY} + \mathbf{V}_{m-k+1}\mathbf{Z}) \qquad (5.8)$$

and observing that the norm of this difference decreases as $\mathbf{U}$ becomes a better approximation of an invariant subspace of $\mathbf{A}$. This is because of the way we defined $\mathbf{Y}$ and $\mathbf{Z}$ in (4.4), which gives an orthogonal decomposition of $\mathbf{U}$ with components in $\mathcal{R}(\mathbf{C})$ and $\mathcal{K}_{m-k+1}((\mathbf{I}-\mathbf{C}\mathbf{C}^*)\mathbf{A},\mathbf{r}_0)$. As $\mathbf{U}$ approaches an invariant subspace, $\|\mathbf{U}-\mathbf{C}\mathbf{Y}\|$ approaches zero. This implies that both $\|\mathbf{Z}\|$ and $\|\mathbf{F}\|$ approach zero. This can also be formulated in terms of a quantity called the *containment gap* [7], which quantifies how close a smaller subspace is to being contained in a larger subspace. It is a generalization of the notion of the *gap* between two subspaces of equal dimension [36].

We must be able to monitor the residuals associated with the shifted systems as well as the other terms in (5.7) which give the relationship between the base residual and shifted residual. This is important because we need to know when the norm of the error introduced by the approximate collinearity condition, namely, $\left\|-\sigma\mathbf{E}\mathbf{F}\left(\widetilde{\mathbf{y}}_m^{(\sigma)}\right)_{1:k}+\mathbf{s}^{(\sigma)}\right\|$, is of the same order of magnitude as $\left\|\widetilde{\beta}_m\mathbf{r}_m\right\|$. At that point, we can no longer guarantee that updating the shifted residual via the approximate collinearity condition will improve the shifted system approximation. Therefore, we should no longer update the shifted system approximation.

Our analysis gives us a way to monitor these quantities. Observe that given σ, $\mathbf{U}$, and $\mathbf{C}$, if we compute $\widetilde{\mathbf{y}}_m^{(\sigma)}$ according to (5.2), then from (5.8), we can compute the product $\mathbf{E}\mathbf{F}\left(\tilde{\mathbf{y}}_m^{(\sigma)}\right)_{1:k}$. Thus, we can keep track of the vector $\mathbf{s}^{(\sigma)}$,which allows us to cheaply construct $\mathbf{r}_m^{(\sigma)}$ according to (5.5). At the expense of one vector of storage, we can store and update $\mathbf{s}^{(\sigma)}$, providing a cheap way to compute $\mathbf{r}_m^{(\sigma)}$.

The vector $\mathbf{s}^{(\sigma)}$ can be easily accumulated. At the beginning of Algorithm 5.1, we compute an initial value of $\mathbf{s}^{(\sigma)}$ according to (5.4). At Line 5 of Algorithm 5.1, we update $\mathbf{s}^{(\sigma)} \leftarrow \mathbf{s}^{(\sigma)} - \sigma\mathbf{U}\mathbf{C}^*\widetilde{\mathbf{r}}^{(\sigma)}$ according to (5.1). At Line 11, we update $\mathbf{s}^{(\sigma)} \leftarrow \mathbf{s}^{(\sigma)} - \sigma\mathbf{E}\mathbf{F}(\widetilde{\mathbf{y}}_m)_{1:k}$ according to (5.5).

It is worth noting that the approximate collinearity condition turns into an exact collinearity condition if we possess a different deflation subspace for the shifted system. Observe that if we write

$$\mathbf{U}^{(\sigma)} = \mathbf{U} - \sigma(\mathbf{A}+\sigma\mathbf{I})^{-1}\mathbf{E}\mathbf{F},$$

then we obtain an exact Arnoldi-like relation

$$(\mathbf{A}+\sigma\mathbf{I})\widehat{\mathbf{V}}_m^{(\sigma)} = \widehat{\mathbf{W}}_{m+1}\widetilde{\mathbf{G}}_m^{(\sigma)}, \tag{5.9}$$

where $\widehat{\mathbf{V}}_m^{(\sigma)} = \begin{bmatrix}\mathbf{U}^{(\sigma)} & \mathbf{V}_{m-k}\end{bmatrix}$. If we select $\mathbf{x}_m^{(\sigma)} \in \mathbf{x}_0^{(\sigma)} + \mathcal{R}(\widehat{\mathbf{V}}_m^{(\sigma)})$ and enforce the collinearity condition $\mathbf{r}_m^{(\sigma)} = \beta_m\mathbf{r}_m$, then we get that (5.2) is the exact collinearity equation which must be solved to obtain the collinear residual. Thus, the singularity of the linear system arising from the approximate collinearity condition in our algorithm is equivalent to the nonexistence of an exactly collinear residual when we construct the approximate solution to (1.4) over a different augmented Krylov subspace. Of course, $\mathbf{U}^{(\sigma)}$ is not available to us in practice; so, its utility here is only as a theoretical tool.

6. Numerical Results.

In the following numerical experiments, we constructed recycled subspaces from harmonic Ritz vectors of the coefficient matrix associated with the base system (1.3) with respect to the augmented subspace. This is by no means the only way to generate a recycled space. A strength of the Recycled GMRES method is the ability to use any recycled subspace. We based our choice on the report in [23] that the use of harmonic Ritz vectors in the recycled space has given

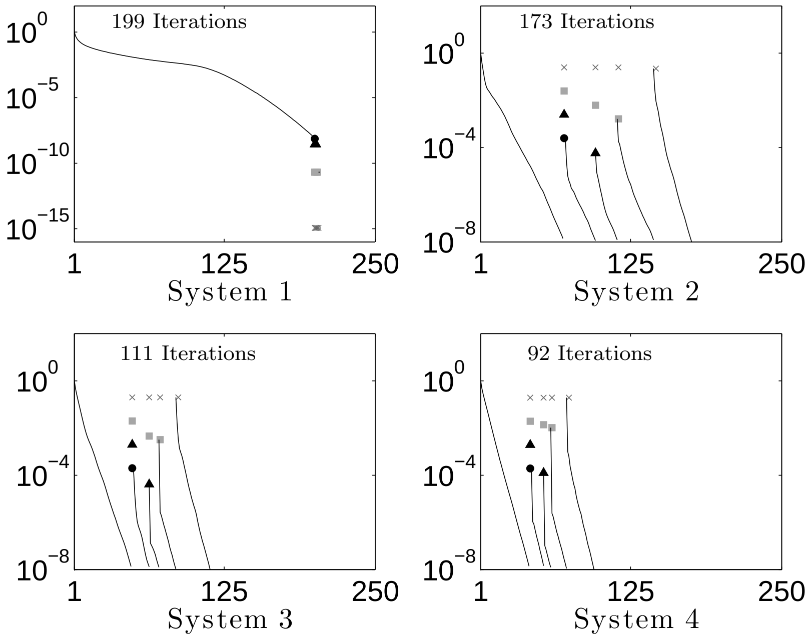

Fig. 6.1. *The performance of Recycled GMRES (RGMRES) for shifted systems with recursion on the number of unconverged shifted systems. We performed simple tests on a sequence of four* 1000×1000 *bidiagonal matrices. For this test,* $m = 100$ *and* $k = 50$. *The first matrix is the bidiagonal matrix used in* [8]. *The other systems are constructed by applying* $\mathcal{O}(1)$ *bidiagonal random perturbations to the first system using the* `sprand()` *Matlab function. The four shifts are* 10^{-2}, 10^{-1}, 1, *and* 10. *For the shifted systems, the residuals are only computed at the end of each cycle, with residual norms represented by the* ***circle, triangle, square,*** *and* ***cross****, respectively. The curves originating from these symbols are the convergence curves for the Recycled GMRES iterations executed for each shift system when that system becomes the base system during a recursive call to the method.*

good results. In the figures, for each recursive call to the algorithm, the solid black line represents the convergence curve for the base system, while the different markers indicate residual norms for the shifted systems at the end of each restart cycle.

It should also be noted when an update to an approximation is calculated according to the approximate collinearity condition, we simply check if the residual norm for each shifted system to see if it has increased or decreased. If it increases, the update is not accepted and all future updates of that approximation are halted in the current cycle. This is based on the assumption that if the residual norm increased, our strategy is no longer effective for that particular shifted residual. This strategy can still be accomplished by accumulating $\mathbf{s}^{(\sigma)}$, as we have already described.

In all experiments, when solving the first system in the sequence, there is no initial recycled subspace. Thus, (4.2) holds at the end of each cycle. This means that solving the first system using Algorithm 5.1 is equivalent to applying shifted GMRES-DR for shifted systems [8]. This can be easily seen in the convergence plots as all residuals are reduced in norm below tolerance when solving the base system. There are no recursive calls to the algorithm. There is one exception; in Figure 6.4, we intentionally chose nonpositive shifts. In the first convergence curve, we see that two recursive calls to the algorithm were necessary for these two shifts.

Our first experiment, in Figure 6.1, illustrates the performance of the Recycled

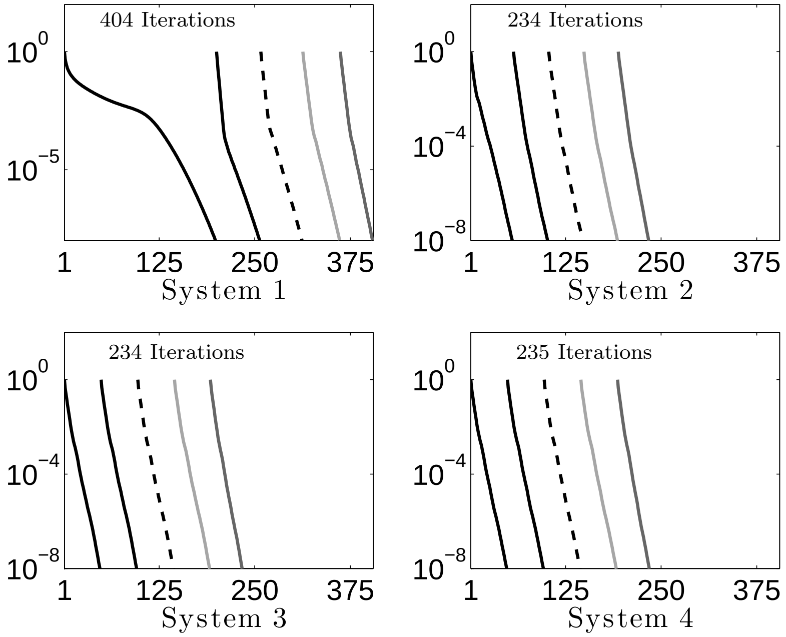

FIG. 6.2. *Convergence curves when we apply Recycled GMRES to each shifted bidiagonal system sequentially. As in the previous experiment, $m = 100$ and $k = 50$.*

GMRES method for shifted linear systems on a sequence of four bidiagonal matrices. The first matrix, $\mathbf{B}_1$, used in [8], is a bidiagonal matrix with the numbers $\{.1, 1, 2, \ldots, 998, 999\}$ on the diagonal and ones on the first superdiagonal and the other matrices are random bidiagonal perturbations of of $\mathbf{B}_1$, with the perturbations having the same bidiagonal structure and having Frobenius norm 1. We see that, as predicted by Corollary 5.1, the amount of residual reduction achieved for the shifted systems is effected by the size of the shift. For the shift $\sigma_1 = 10^{-2}$, the relative residual is reduced to $\mathcal{O}(10^{-4})$ during the solution of the base system while for $\sigma_4 = 10$, the relative residual is only reduced to $\mathcal{O}(10^{-1})$. This experiment is more for illustrative purposes than to demonstrate superior performance. After convergence for the base system, we take one of the shifted systems as our new base system and reapply the algorithm for the smaller family of systems. For comparison, we present in Figure 6.2 the convergence curves if we simply apply Recycled GMRES to each shifted system sequentially.

For our second and third experiments, we test two sequences of QCD matrices from the University of Florida sparse matrix collection [9]. In the second experiment, we work with six 3072×3072 sample matrices (called $\mathbf{D}_1$ through $\mathbf{D}_6$) with filename prefix `conf5.0-00l4x4`. We can construct the coefficient matrix $\mathbf{A}_i = \mathbf{I} - \kappa^{(i)}\mathbf{D}_i$ where $\kappa^{(i)}$ is a parameter associated to the QCD problem. For each matrix, there exists some critical value $\kappa_c^{(i)}$ such that for $0 \leq \kappa^{(i)} < \kappa_c^{(i)}$, $\mathbf{A}_i$ is a real-positive matrix. Equivalently, for each $\mathbf{A}_i$, we can write $\mathbf{A}_i = \frac{1}{\kappa^{(i)}}\mathbf{I} - \mathbf{D}_i$ where $\frac{1}{\kappa_c^{(i)}} < \frac{1}{\kappa^{(i)}} < \infty$, and we can scale any right-hand-side so that we are solving the same problem. For each $\mathbf{D}_i$, $\kappa_c^{(i)}$ is included with the matrix, and in these experiments, all are in the interval $[0.20, 0.22]$. Frequently in QCD computations, we wish to solve with multiple parameters, and we can solve all these systems simultaneously using an algorithm for

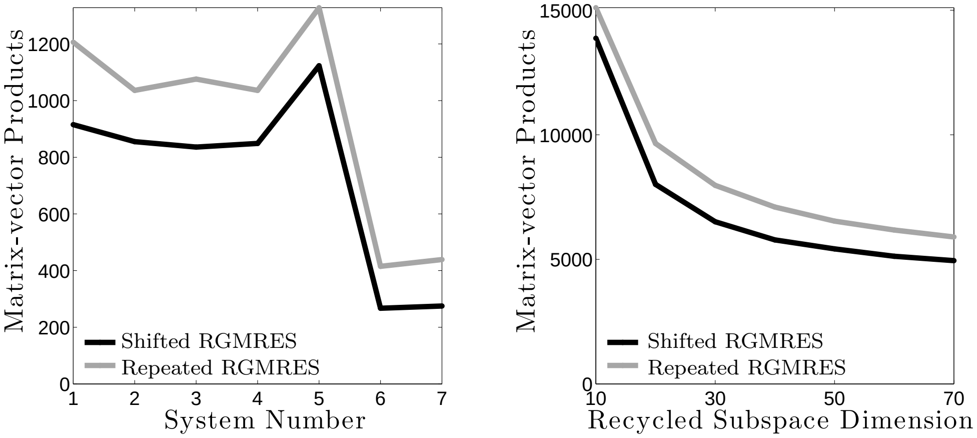

Fig. 6.3. *The performance of Recycled GMRES for shifted systems on a sequence of seven small Wilson fermion matrices where for each matrix, we solve a linear system with the base system and those associated to the shifts,* .001 , .002, .003, $-.6$, *and* $-.5$. *In the left figure, we illustrate the performance of RMGRES(*100,50*) for shifted systems as compared to repeated applications of RGMRES(*100,50*). In the figure on the right, we compare performance for different size recycled subspaces with* $m = 100$ *fixed.*

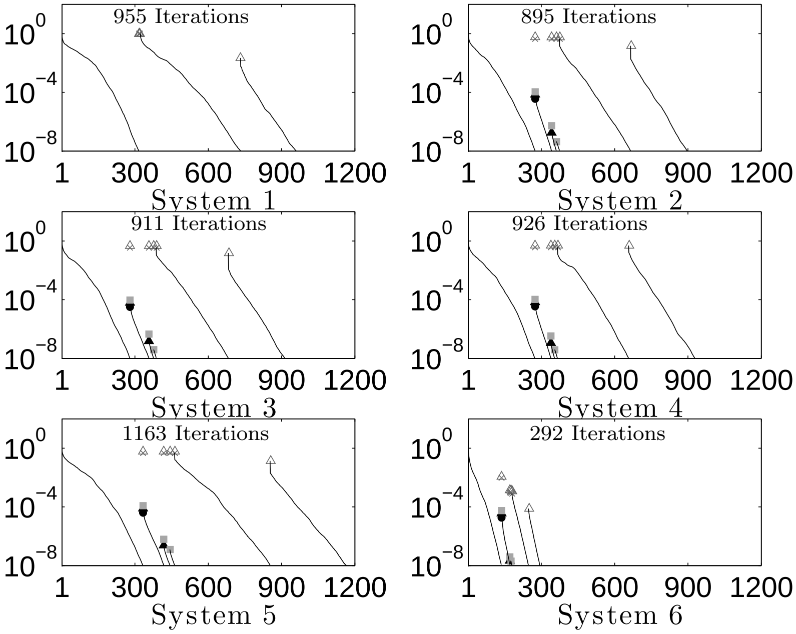

Fig. 6.4. *Convergence curves from the same experiment as in Figure 6.3 but only for the first six systems. We again use* $m = 100$ *and* $k = 50$. *Observe that two of the shifted systems (corresponding to the negative shifts) require more work than the others for each base system* ***including*** *the first system, in which we started with no recycled subspace. These are the two shifts for which shifted GMRES (or shifted GMRES-DR) would not converge. Notice that in this case, we are still able to converge by applying Recycled GMRES at the end.*

shifted linear systems.

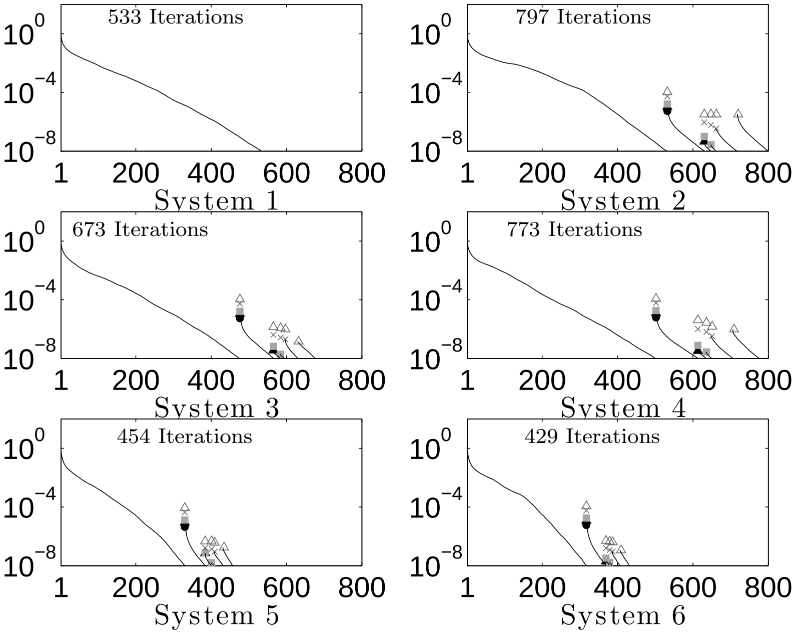

FIG. 6.5. *Convergence curves for another sequence of six Wilson fermion matrices, of size* 49152×49152 *with* $m = 100$ *and* $k = 50$.

We chose $\{.001, .002, .003, -.6, -.5\}$ as our family of shifts. Observe that by the definition of $\mathbf{A}_i$ and $\kappa_c^{(i)}$, the two shifted coefficient matrices associated with the two negative shifts are not real-positive. These are not physically relevant for QCD computations. We chose negative shifts merely to demonstrate the robustness of the algorithm. In this experiment, GMRES for shifted systems was unable to produce approximations for long sequences of iterations (due to singularity of the augmented collinearity matrix). Since the shifted GMRES method did not converge for some systems, its performance was not included in the figure. However, as we have noted, it is not difficult to modify this algorithm to gracefully handle this situation by applying restarted GMRES to any unconverged shifted systems at the end of the process. We compared with another strategy, repeated applications of Recycled GMRES [23] for the base and shifted system. In Figure 6.3 we present the matrix-vector product counts for each system for a particular recycled subspace dimension as well as the totals over seven systems for various recycled subspace dimensions. We see that our method is able to produce a 20% reduction in the number of matrix-vector products needed to solve these systems, when compared to repeated applications of Recycled GMRES. In Figure 6.4, we present the convergence curves for the first six QCD matrices.

In the third experiment, we worked with another sequence of QCD matrices from Matrix Market [1] with filename prefixes `conf5.4` and `conf6.0`. These matrices of of size 49152×49152. We used the critical $\kappa_c^{(i)}$ to construct our system matrices as in the second experiment, and we choose the shifts $\{.001, .002, .003, .01, .02\}$ as in the second experiment. In Figure 6.5, we see the convergence of our algorithm for these systems.

TABLE 6.1

A comparison, in terms of iteration counts of the shifted GMRES algorithm (SGMRES) with the Recycled GMRES algorithm for shifted systems (RGMRES) in terms of iteration count for different cycle lengths m. The results presented are the total iterations for solving a sequence of eleven QCD systems from [20].

m	k	SGMRES(m)	RGMRES($m-k$,k)	ratio
25	12	4297	3880	0.90
50	25	3284	2980	0.91
75	37	3108	2816	0.91
100	50	3028	2697	0.89
125	67	3058	2612	0.85
150	75	2958	2546	0.86
175	87	2962	2499	0.84
200	100	2947	2458	0.83
225	112	2860	2410	0.84

In the fourth experiment, we work with a sequence of eleven QCD matrices obtained from [20]. These matrices were delivered already shifted to be positive-real. As in the previous experiment, they are also of size 49152 × 49152. In Table 6.1, we illustrate the performance of the algorithm when the total dimension of the augmented space increases and compared this performance to an equivalent instance of the shifted GMRES algorithm. What this means is that we compared the performance of shifted GMRES with cycle length m versus our algorithm with an $k = \lfloor m/2 \rfloor$ dimension deflation space and $m-k$ cycle length. In this experiment, there are two shifts, $\{.8, .81\}$. Here we see the potential benefits that our algorithm can yield as the deflation dimension increases. We recognize that in some cases, such as in the last example, the gain as compared with RGMRES is relatively small, and that a larger number of shifts (or different values for the shifts) may not give this advantage. This follows from the fact that each shift incurs an additional recursive call to the algorithm and additional iterations. There is no such increase for shifted GMRES. Therefore, for sufficiently large number of shifts, shifted GMRES will have an advantage. Which method will perform better depends on several factors including the number of shifts (as we just mentioned), the magnitude of the shifts, the size of deflation space, and the deflation space selection technique. What we have shown is that for certain problems, the recycling strategy may be worth considering.

7. Conclusions.

We have shown that the attractive idea of efficiently combining techniques to simultaneously solve a family of shifted linear systems with subspace recycling techniques presents difficulties which cannot be easily surmounted. As an alternative, we present a technique that does as good as we can hope for using the subspace recycling framework. While this technique does not simultaneously solve all systems, it allows us to cheaply compute improvements to the approximate solution to the shifted systems while computing the minimum residual solution over an augmented Krylov subspace. When the base system has been solved, the method can be called recursively on the remaining systems, where one unconverged shifted system becomes the new base system. Numerical results demonstrates that the method can be effective.

Acknowledgments. We would like to thank David Day and Michael Parks for engaging in fruitful discussions with the first author. We also thank Andreas Frommer

for his extensive comments, including the suggestion that Algorithm 5.1 is amenable to recursion.

Appendix A. Constructing Multiple Deflation Spaces.

We now return to the question of constructing multiple deflation spaces.

Given a matrix $\widetilde{\mathbf{U}} \in \mathbb{C}^{n\times k}$ with columns spanning our recycled space, how do we compute $\mathbf{U}$ and $\mathbf{U}^{(\sigma)}$ such that both

$$\mathbf{A}\mathbf{U} = \mathbf{C} \quad \text{and} \quad (\mathbf{A} + \sigma\mathbf{I})\mathbf{U}^{(\sigma)} = \mathbf{C} \tag{A.1}$$

hold, with $\mathbf{C}^*\mathbf{C} = \mathbf{I}_k$? In GCRODR, we compute $\mathbf{A}\widetilde{\mathbf{U}} = \widetilde{\mathbf{C}}$. Then we compute the QR-factorization $\widetilde{\mathbf{C}} = \mathbf{C}\mathbf{R}$ and let $\mathbf{U} = \widetilde{\mathbf{U}}\mathbf{R}^{-1}$. One way to interpret this process is to consider that to solve one system, we need one recycled space satisfying (2.8) which costs k matrix-vector products plus one basis orthogonalization to acquire.

Now suppose we have two coefficient matrices $\mathbf{A}$ and $(\mathbf{A} + \sigma\mathbf{I})$. If we have $\mathbf{U}$ such that $\mathbf{A}\mathbf{U} = \mathbf{C}$, there is no efficient way to acquire $\mathbf{U}^{(\sigma)}$. We cannot compute the image of $\widetilde{\mathbf{U}}$ under $\mathbf{A}$ and $\mathbf{A}+\sigma\mathbf{I}$ separately because this would not yield one matrix $\mathbf{C}$. To get a pair of matrices $\mathbf{U}$ and $\mathbf{U}^{(\sigma)}$ satisfying (A.1), we premultiply $\widetilde{\mathbf{U}}$ by both $\mathbf{A}$ and $\mathbf{A} + \sigma\mathbf{I}$. In other words, let $\widetilde{\mathbf{C}} = \mathbf{A}(\mathbf{A} + \sigma\mathbf{I})\widetilde{\mathbf{U}}$ and compute the QR-factorization $\widetilde{\mathbf{C}} = \mathbf{C}\mathbf{R}$. Then we have $\mathbf{U} = (\mathbf{A} + \sigma\mathbf{I})\widetilde{\mathbf{U}}\mathbf{R}^{-1}$ and $\mathbf{U}^{(\sigma)} = \mathbf{A}\widetilde{\mathbf{U}}\mathbf{R}^{-1}$. Since matrices differing by multiples of the identity commute, we have $\mathbf{A}\mathbf{U} = (\mathbf{A} + \sigma\mathbf{I})\mathbf{U}^{(\sigma)} = \mathbf{C}$. This easily can be generalized to any number of shifts. Given a sequence of shifts $\left\{\sigma^i\right\}_{i=1}^{s+1} \subset \mathbb{C}$, where we consider the base matrix as a system with shift $\sigma_1 = 0$, the

Algorithm A.1: Computing a Family of Recycled Spaces for Multiple Shifted Linear Systems

```
Input  : $\mathbf{A} \in \mathbb{C}^{n\times n}$, $\{\sigma_i\}_{i=1}^{s+1} \subset \mathbb{C}$, $\widetilde{\mathbf{U}} \in \mathbb{C}^{n\times k}$
Output: $\mathbf{C} \in \mathbb{C}^{n\times k}$ such that $\mathbf{C}^*\mathbf{C} = \mathbf{I}_k$, $\left\{\mathbf{U}^{(\sigma_i)}\right\}_{i=1}^{s+1}$ such that
         $(\mathbf{A} + \sigma_i\mathbf{I})\mathbf{U}^{(\sigma_i)} = \mathbf{C}$ for all $i$
for $i = 1$ to $s + 1$ do
    $\mathbf{U}^{(\sigma_i)} = \widetilde{\mathbf{U}}$
for $i = 1$ to $s$ do
    $\mathbf{U}^{(\sigma_{s+1})} \leftarrow \mathbf{A}\mathbf{U}^{(\sigma_{s+1})} + \sigma_i\mathbf{U}^{(\sigma_{s+1})}$
    Compute the QR-factorization $\mathbf{U}^{(\sigma_{s+1})} = \mathbf{Q}\mathbf{R}$
    $\mathbf{U}^{(\sigma_{s+1})} \leftarrow \mathbf{Q}$
    for $j = 1$ to $s$ do
        $\widehat{\mathbf{U}} \leftarrow \mathbf{U}^{(\sigma_j)}\mathbf{R}^{-1}$
        $k_1 \leftarrow (j - 1 \mod s + 1) + (1 - \lceil \frac{j-1}{s+1} \rceil)(s + 1)$
        $k_2 \leftarrow (j + 1 \mod s + 1) + \lfloor \frac{j+1}{s+1} \rfloor (s + 1)$
        $\mathbf{U}^{(\sigma_j)} \leftarrow \mathbf{U}^{(\sigma_{k_1})} + (\sigma_{k_2} - \sigma_j)\widehat{\mathbf{U}}$
$\mathbf{C} = \mathbf{A}\mathbf{U}^{(\sigma_{s+1})} + \sigma_{s+1}\mathbf{U}^{(\sigma_{s+1})}$
Compute the QR-factorization $\mathbf{C} = \mathbf{Q}\mathbf{R}$
$\mathbf{C} \leftarrow \mathbf{Q}$
for $i = 1$ to $s + 1$ do
    $\mathbf{U}^{(\sigma_i)} \leftarrow \mathbf{U}^{(\sigma_i)}\mathbf{R}^{-1}$
```

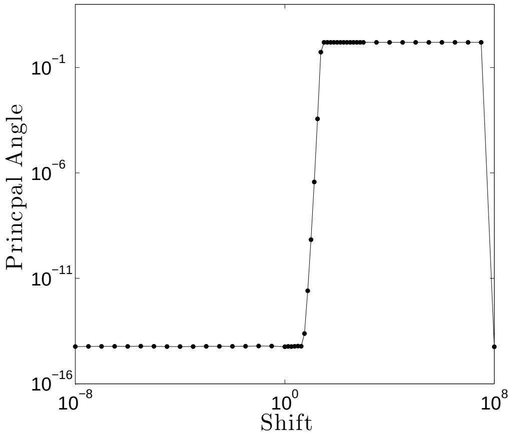

FIG. A.1. *The performance of Algorithm A.1 for a collection of shifts taken from the interval* $\left[10^{-8}, 10^{8}\right]$*. For this figure, both axes are on a logarithmic scale. We compute the largest principal angle between* $\mathcal{R}(\mathbf{C})$ *and* $\mathcal{R}((\mathbf{A}{+}\sigma\mathbf{I})\mathbf{U}^{(\sigma)})$ *to examine the accuracy.* ***Observe*** *that there is an increase in accuracy for the right-most shift,* $\sigma = 10^{8}$*. The subspace associated to* $\sigma = 10^{8}$ *corresponds to* $\mathbf{U}^{(\sigma_{s+1})}$ *in Algorithm A.1. The computation of* $\mathbf{U}^{(\sigma_{s+1})}$ *is done directly rather than by using Proposition A.1.*

matrices

$$\mathbf{U}^{(\sigma_j)} = \left[\prod_{i=1,i\neq j}^{s+1} (\mathbf{A} + \sigma_i)\right] \widetilde{\mathbf{U}}\mathbf{R}^{-1} \quad \text{and} \quad \mathbf{C} = \left[\prod_{i=1}^{s+1} (\mathbf{A} + \sigma_i)\right] \widetilde{\mathbf{U}}\mathbf{R}^{-1}$$

satisfy our requirements with $\widetilde{\mathbf{C}} = \mathbf{C}\mathbf{R}$ being the QR-factorization of $\widetilde{\mathbf{C}}$. Since matrices differing by a multiple of the identity commute, we have that $(\mathbf{A} + \sigma_i\mathbf{I})\mathbf{U}^{(\sigma_i)} = \mathbf{C}$ for all i. This strategy has clear stability issues which must be addressed. Furthermore, we wish to execute as few calls to the operator $\mathbf{A}$ as possible. Therefore, we propose a more stable implementation of this procedure.

Though these matrices do commute, once we choose a particular multiplication order, e.g., if $\mathbf{C} = \mathbf{A}(\mathbf{A} + \sigma\mathbf{I})\widetilde{\mathbf{U}}\mathbf{R}^{-1}$, we must proceed intelligently to recover $\mathbf{U}^{(\sigma)} = \mathbf{A}\widetilde{\mathbf{U}}\mathbf{R}^{-1}$ without doing additional matrix vector products. To deal with this issue, we first introduce a straightforward proposition whose proof follows by simple algebraic manipulation.

PROPOSITION A.1. *Let* $\widehat{\mathbf{U}} = (\mathbf{A} + \sigma_1\mathbf{I})\widetilde{\mathbf{U}}$*. Then* $(\mathbf{A} + \sigma_2\mathbf{I})\widetilde{\mathbf{U}} = \widehat{\mathbf{U}} + (\sigma_2 - \sigma_1)\widetilde{\mathbf{U}}$*. Furthermore, let* $\widehat{\mathbf{U}} = \mathbf{C}\mathbf{R}$ *be the QR-factorization. Then*

$$(\mathbf{A} + \sigma_2\mathbf{I})\widetilde{\mathbf{U}}\mathbf{R}^{-1} \;=\; \mathbf{C} \;+\; (\sigma_2 - \sigma_1)\widetilde{\mathbf{U}}\mathbf{R}^{-1}.$$

For clarity, we present a small example to motivate a general algorithm for this process. Suppose we have a family of three shifted systems $\mathbf{A} + \sigma_1\mathbf{I}$, $\mathbf{A} + \sigma_2\mathbf{I}$, and $\mathbf{A} + \sigma_3\mathbf{I}$. Our goal is, given $\widehat{\mathbf{U}} \in \mathbb{C}^{n\times k}$, to generate $\mathbf{U}^{(\sigma_1)}, \mathbf{U}^{(\sigma_2)}, \mathbf{U}^{(\sigma_3)} \in \mathbb{C}^{n\times k}$ such that for some $\mathbf{C} \in \mathbb{C}^{n\times k}$ such that $\mathbf{C}^*\mathbf{C} = \mathbf{I}_k$, we have

$$(\mathbf{A} + \sigma_1\mathbf{I})\mathbf{U}^{(\sigma_1)} = \mathbf{C} \quad \text{and} \quad (\mathbf{A} + \sigma_2\mathbf{I})\mathbf{U}^{(\sigma_2)} = \mathbf{C} \quad \text{and} \quad (\mathbf{A} + \sigma_3\mathbf{I})\mathbf{U}^{(\sigma_3)} = \mathbf{C}.$$

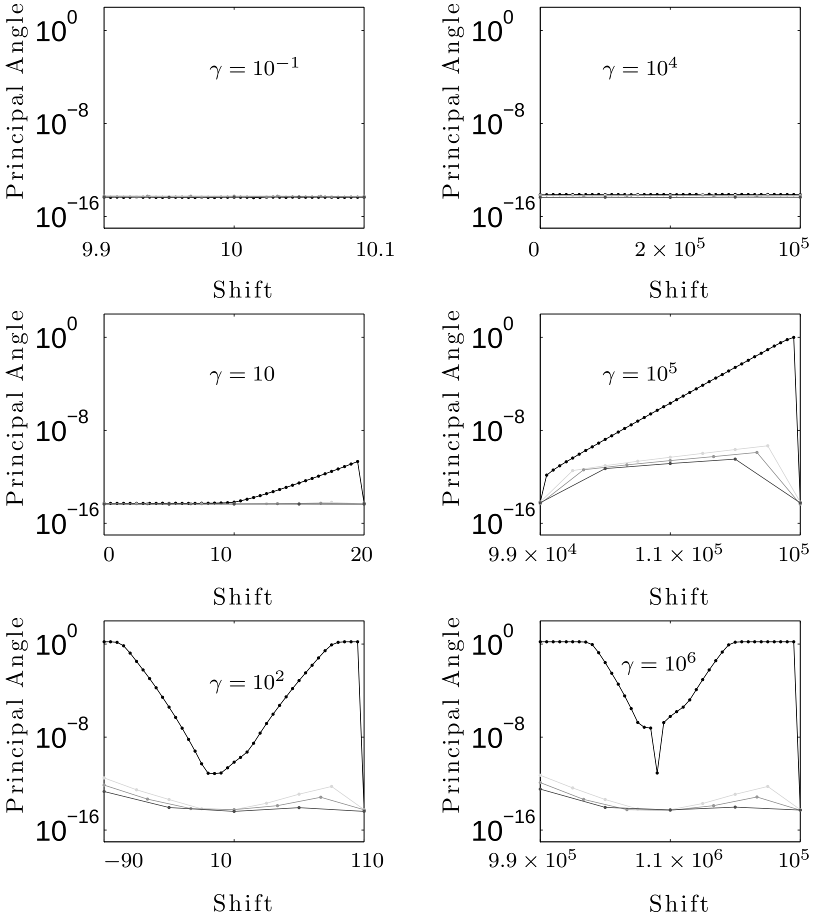

FIG. A.2. *The performance of Algorithm A.1 for collections of shifts. Different lines depict the algorithm's performance for different numbers of shifts. Line darkness increases when more shifts are used (meaning the algorithm requires more iterations). We compute the largest principal angle between* $\mathcal{R}(\mathbf{C})$ *and* $\mathcal{R}((\mathbf{A} + \sigma\mathbf{I})\mathbf{U}^{(\sigma)})$ *to examine the accuracy. In the figures in the left column, shifts come from the interval* $[10 - \gamma, 10 + \gamma]$. *In the right column, shifts come from the interval* $\left[10^5 - \gamma, 10^5 + \gamma\right]$. *Observe that the Algorithm A.1 performs well for larger values of* γ *when the center is located at* 10^5.

Furthermore, we want to do this using only $3k$ matrix-vector products (or three block k matrix-vector products) and k-dimensional basis orthogonalizations. This is a three-step process, and at each step, we will compute k matrix-vector products (or one block k matrix-vector product) and then apply Proposition A.1 to implicitly compute other matrix-vector products needed in that step. To begin, let

$$\mathbf{U}_0^{(\sigma_1)} = \mathbf{U}_0^{(\sigma_2)} = \mathbf{U}_0^{(\sigma_3)} = \widehat{\mathbf{U}}.$$

We want to

1. Compute $\widehat{\mathbf{U}}_1^{(\sigma_1)} = (\mathbf{A} + \sigma_2\mathbf{I})\mathbf{U}_0^{(\sigma_1)}$, $\widehat{\mathbf{U}}_1^{(\sigma_2)} = (\mathbf{A} + \sigma_3\mathbf{I})\mathbf{U}_0^{(\sigma_2)}$, and $\widehat{\mathbf{U}}_1^{(\sigma_3)} = (\mathbf{A} + \sigma_1\mathbf{I})\mathbf{U}_0^{(\sigma_3)}$
2. Compute the QR-factorization $\widehat{\mathbf{U}}_1^{(\sigma_3)} = \mathbf{U}_1^{(\sigma_3)}\mathbf{R}_1$
3. Compute $\mathbf{U}_1^{(\sigma_1)} = \widehat{\mathbf{U}}_1^{(\sigma_1)}\mathbf{R}_1^{-1}$ and $\mathbf{U}_1^{(\sigma_2)} = \widehat{\mathbf{U}}_1^{(\sigma_2)}\mathbf{R}_1^{-1}$.

However, we do not need to do these computations explicitly. If we compute $\widehat{\mathbf{U}}_1^{(\sigma_3)} = (\mathbf{A} + \sigma_1\mathbf{I})\mathbf{U}_0^{(\sigma_2)}$ and the QR-factorization $\widehat{\mathbf{U}}_1^{(\sigma_3)} = \mathbf{U}_1^{(\sigma_3)}\mathbf{R}_1$, then by Proposition A.1, we know that

$$\mathbf{U}_1^{(\sigma_1)} = \mathbf{U}_1^{(\sigma_3)} + (\sigma_2 - \sigma_1)\mathbf{U}_0^{(\sigma_1)}\mathbf{R}_1^{-1} \tag{A.2}$$

which then implies that, again by Proposition A.1,

$$\mathbf{U}_1^{(\sigma_2)} = \mathbf{U}_1^{(\sigma_1)} + (\sigma_3 - \sigma_2)\mathbf{U}_0^{(\sigma_2)}\mathbf{R}_1^{-1}.$$

For the next step, explicitly, we want to

1. Compute $\widehat{\mathbf{U}}_2^{(\sigma_1)} = (\mathbf{A} + \sigma_3\mathbf{I})\mathbf{U}_1^{(\sigma_1)}$, $\widehat{\mathbf{U}}_2^{(\sigma_2)} = (\mathbf{A} + \sigma_1\mathbf{I})\mathbf{U}_1^{(\sigma_2)}$, and $\widehat{\mathbf{U}}_2^{(\sigma_3)} = (\mathbf{A} + \sigma_2\mathbf{I})\mathbf{U}_1^{(\sigma_2)}$
2. Compute the QR-factorization $\widehat{\mathbf{U}}_2^{(\sigma_3)} = \mathbf{U}_2^{(\sigma_3)}\mathbf{R}_2$
3. Compute $\mathbf{U}_2^{(\sigma_1)} = \widehat{\mathbf{U}}_2^{(\sigma_1)}\mathbf{R}_2^{-1}$ and $\mathbf{U}_2^{(\sigma_2)} = \widehat{\mathbf{U}}_2^{(\sigma_2)}\mathbf{R}_2^{-1}$.

Again, however, we do not need to do all these computations explicitly. If we compute $\widehat{\mathbf{U}}_2^{(\sigma_3)} = (\mathbf{A} + \sigma_2\mathbf{I})\mathbf{U}_1^{(\sigma_2)}$ and the QR-factorization $\widehat{\mathbf{U}}_2^{(\sigma_3)} = \mathbf{U}_2^{(\sigma_3)}\mathbf{R}_2$, then we have

$$\begin{aligned}\widehat{\mathbf{U}}_2^{(\sigma_1)} &= (\mathbf{A} + \sigma_3\mathbf{I})\mathbf{U}_1^{(\sigma_1)} \qquad \text{(A.3)}\\ &= (\mathbf{A} + \sigma_3\mathbf{I})(\mathbf{U}_1^{(\sigma_3)} + (\sigma_2 - \sigma_1)\mathbf{U}_0^{(\sigma_1)}\mathbf{R}_1^{-1})\\ &= (\mathbf{A} + \sigma_2\mathbf{I})\mathbf{U}_1^{(\sigma_3)} + (\sigma_3 - \sigma_2)\mathbf{U}_1^{(\sigma_3)} + (\sigma_2 - \sigma_1)(\mathbf{A} + \sigma_3\mathbf{I})\mathbf{U}_0^{(\sigma_1)}\mathbf{R}_1^{-1}\\ &= \widehat{\mathbf{U}}_2^{(\sigma_3)} + (\sigma_3 - \sigma_2)\mathbf{U}_1^{(\sigma_3)} + (\sigma_2 - \sigma_1)(\mathbf{A} + \sigma_3\mathbf{I})\mathbf{U}_0^{(\sigma_1)}\mathbf{R}_1^{-1}.\end{aligned}$$

By Proposition A.1,

$$(\mathbf{A} + \sigma_3\mathbf{I})\mathbf{U}_0^{(\sigma_1)}\mathbf{R}_1^{-1} = \mathbf{U}_1^{(\sigma_3)} + (\sigma_3 - \sigma_1)\mathbf{U}_0^{(\sigma_1)}\mathbf{R}_1^{-1},$$

yielding

$$\begin{aligned}\widehat{\mathbf{U}}_2^{(\sigma_1)} &= \widehat{\mathbf{U}}_2^{(\sigma_3)} + (\sigma_3 - \sigma_2)\mathbf{U}_1^{(\sigma_3)} + (\sigma_2 - \sigma_1)(\mathbf{U}_1^{(\sigma_3)} + (\sigma_3 - \sigma_1)\mathbf{U}_0^{(\sigma_1)}\mathbf{R}_1^{-1})\\ &= \widehat{\mathbf{U}}_2^{(\sigma_3)} + (\sigma_3 - \sigma_2)\mathbf{U}_1^{(\sigma_3)} + (\sigma_2 - \sigma_1)\mathbf{U}_1^{(\sigma_3)} + (\sigma_2 - \sigma_1)(\sigma_3 - \sigma_1)\mathbf{U}_0^{(\sigma_1)}\mathbf{R}_1^{-1}\\ &= \widehat{\mathbf{U}}_2^{(\sigma_3)} + (\sigma_3 - \sigma_2)\mathbf{U}_1^{(\sigma_3)} + (\sigma_2 - \sigma_1)\mathbf{U}_1^{(\sigma_3)} + (\sigma_2 - \sigma_1)(\sigma_3 - \sigma_1)\mathbf{U}_0^{(\sigma_1)}\mathbf{R}_1^{-1}\\ &= \widehat{\mathbf{U}}_2^{(\sigma_3)} + (\sigma_3 - \sigma_1)\mathbf{U}_1^{(\sigma_3)} + (\sigma_2 - \sigma_1)(\sigma_3 - \sigma_1)\mathbf{U}_0^{(\sigma_1)}\mathbf{R}_1^{-1}\\ &= \widehat{\mathbf{U}}_2^{(\sigma_3)} + (\sigma_3 - \sigma_1)(\mathbf{U}_1^{(\sigma_3)} + (\sigma_2 - \sigma_1)\mathbf{U}_0^{(\sigma_1)}\mathbf{R}_1^{-1}).\end{aligned}$$

Finally, by (A.2), we have that

$$\widehat{\mathbf{U}}_2^{(\sigma_1)} = \widehat{\mathbf{U}}_2^{(\sigma_3)} + (\sigma_3 - \sigma_1)\mathbf{U}_1^{(\sigma_1)},$$

which, when we post multiply by $\mathbf{R}_2^{-1}$, gives us

$$\mathbf{U}_2^{(\sigma_1)} = \mathbf{U}_2^{(\sigma_3)} + (\sigma_3 - \sigma_1)\mathbf{U}_1^{(\sigma_1)}\mathbf{R}_2^{-1}.$$

A similar derivation shows us that

$$\mathbf{U}_2^{(\sigma_2)} = \mathbf{U}_2^{(\sigma_1)} + (\sigma_1 - \sigma_2)\mathbf{U}_1^{(\sigma_2)}\mathbf{R}_2^{-1}.$$

From here, we can compute the QR-factorization $(\mathbf{A} + \sigma_3\mathbf{I})\mathbf{U}_2^{(\sigma_3)} = \mathbf{C}\mathbf{R}_3$ and let

$$\mathbf{U}^{(\sigma_1)} = \mathbf{U}_2^{(\sigma_1)}\mathbf{R}_3^{-1} \quad \text{and} \quad \mathbf{U}^{(\sigma_2)} = \mathbf{U}_2^{(\sigma_2)}\mathbf{R}_3^{-1} \quad \text{and} \quad \mathbf{U}^{(\sigma_3)} = \mathbf{U}_2^{(\sigma_3)}\mathbf{R}_3^{-1}.$$

For the general case, in which we have $s + 1$ shifts, this suggests an $(s+1)$-step algorithm for the computation $\mathbf{U}^{(\sigma_i)}$ for all i and $\mathbf{C}$ such that $\mathbf{C}^*\mathbf{C} = \mathbf{I}_k$. We begin with some notation. For any shift σ_i, $\mathbf{U}^{(\sigma_i)}$ is such that $(\mathbf{A} + \sigma_i\mathbf{I})\mathbf{U}^{(\sigma_i)} = \mathbf{C}$. At the jth step, we compute k matrix-vector products and generate an intermediate matrix $\mathbf{U}_j^{(\sigma_i)}$ for each shift, and we denote the initial recycled space associated to σ_i, $\mathbf{U}_0^{(\sigma_i)} = \widetilde{\mathbf{U}}$. At step j, we compute the product

$$\widehat{\mathbf{U}}_j^{(\sigma_i)} \;=\; (\mathbf{A} + \sigma_{\rho_j}\mathbf{I})\mathbf{U}_{j-1}^{(\sigma_i)},$$

where $\rho_j = (j+1 \mod s+1) + \lfloor \frac{j+1}{s+1} \rfloor (s+1)$. This formula for ρ_j allows us to cyclically index through all the shifts using arithmetic on the group $\mathbb{Z}_{s+1} = \{1, 2, \ldots, s+1\}$. For stability, we need to maintain orthogonality of one of the bases. Therefore, without loss of generality, we compute the QR-factorization $\widehat{\mathbf{U}}_j^{(\sigma_{s+1})} \;=\; \mathbf{U}_j^{(\sigma_{s+1})}\mathbf{R}_j$ and update $\mathbf{U}_j^{(\sigma_i)} = \widehat{\mathbf{U}}_j^{(\sigma_i)}\mathbf{R}_j^{-1}$ for each $i \neq s+1$. In each step of this process, we only compute k matrix-vector products explicitly (those associated to σ_{s+1}). The others are computed using Proposition A.1. At the end of the process, we have computed sk matrix-vector products (or s block k matrix-vector products). We finish by computing $\widehat{\mathbf{C}} = (\mathbf{A} + \sigma_{s+1}\mathbf{I})\mathbf{U}_s^{(\sigma_{s+1})}$ and one final QR-factorization $\widehat{\mathbf{C}} = \mathbf{C}\mathbf{R}$. We then update $\mathbf{U}^{(\sigma_i)} = \mathbf{U}_s^{(\sigma_i)}\mathbf{R}^{-1}$ for all i. We present these steps more precisely in Algorithm A.1.

In Figure A.1, we demonstrate the performance of Algorithm A.1 for a set of shifts ranging from 10^{-8} to 10^{8}. We see that for shifts which are $\mathcal{O}(1)$ or smaller in magnitude, the algorithm performs quite well. However, for larger shifts, we see a dramatic loss of accuracy, where we define accuracy to mean the largest principal angle between $\mathcal{R}((\mathbf{A} + \sigma\mathbf{I})\mathbf{U}^{(\sigma)})$ and $\mathcal{R}(\mathbf{C})$. In the computation of these subspaces, we must choose one shift for which we will compute $\mathbf{U}^{(\sigma)}$ directly. Then we use Proposition A.1 to implicitly do the other computations. We chose the rightmost shift to be the one for which we explicitly do the computations. Thus, we get a much smaller principal angles for the subspace associated to the last shift. It's not affected by any numerical errors are causing the algorithm's performance degradation for the other shifts. Further experiments show that it is not just the magnitude of the shifts that matter. Experimentally, we observe that three other criteria on the set of shifts also seem to affect the stability of the algorithm. For a collection of real shifts, let $\{\sigma_i\}_{i=1}^{s} \subset [a, b]$ with $a < b$ where a and b are the largest and smallest such numbers for which the interval contains the shifts, respectively. We have observed that $|b - a|$ affects stability. However, this effect is relative to the distance of the set of shifts from the origin, where we define distance as $|b + a|/2$, the distance of the midpoint from the origin. The further the distance of the set from the origin, the larger $|b - a|$ can be before instability onsets. This is demonstrated by experiments shown in Figure A.2. What determines the performance of Algorithm A.1 is how closely clustered the shifts are, where "close" is defined relative to the distance of the midpoint from the origin. Again, we see the drop in principal angle magnitude for the rightmost shift.

Suppose that the initial recycled space $\widetilde{\mathbf{U}}$ is an approximate invariant subspace of $\mathbf{A}$ associated to small-magnitude eigenvalues. The process described in Algorithm A.1 might yield a matrix $\mathbf{C}$ with columns spanning a new approximate invariant subspace. This new subspace might not be associated to small magnitude eigenvalues, though. Because $\mathbf{C}$ is constructed implicitly by premultiplying $\widetilde{\mathbf{U}}$ by different members of the family of shifted matrices, it is not clear what eigenspace $\mathcal{R}(\mathbf{C})$ might approximate.

REFERENCES

[1] *Matrix Market website*, 2011. `http://math.nist.gov/MatrixMarket/`.

[2] Mian Ilyas Ahmad, Daniel B. Szyld, and Martin B. van Gijzen, *Preconditioned multishift BiCG for $\mathcal{H}_2$-optimal model reduction*, Research Report 12-06-15, Department of Mathematics, Temple University, June 2012.

[3] Allison H. Baker, Elizabeth R. Jessup, and Thomas Manteuffel, *A technique for accelerating the convergence of restarted GMRES*, SIAM Journal on Matrix Analysis and Applications, 26 (2005), pp. 962–984.

[4] Stephen L. Campbell, Ilse C. F. Ipsen, C. Tim Kelley, and Carl D. Meyer, *GMRES and the minimal polynomial*, BIT, 36 (1996), pp. 664–675.

[5] Maria R. Celis, John E. Dennis, and Richard A. Tapia, *A trust region strategy for nonlinear equality constrained optimization*, in Numerical optimization, 1984 (Boulder, Colo., 1984), Paul T. Boggs, Richard H. Byrd, and Robert B. Schnabel, eds., SIAM, Philadelphia, PA, 1985, pp. 71–82.

[6] Tony F. Chan and Wing L. Wan, *Analysis of projection methods for solving linear systems with multiple right-hand sides*, SIAM Journal on Scientific Computing, 18 (1997), pp. 1698–1721.

[7] Francoise Chatelin, *Eigenvalues of Matrices*, Wiley, Chichester, UK, 1993.

[8] Dean Darnell, Ronald B. Morgan, and Walter Wilcox, *Deflated GMRES for systems with multiple shifts and multiple right-hand sides*, Linear Algebra and its Applications, 429 (2008), pp. 2415–2434.

[9] Timothy A. Davis and Yifan Hu, *The University of Florida sparse matrix collection*, ACM Trans. Math. Softw., 38 (2011), pp. 1:1–1:25.

[10] Michael Eiermann, Oliver G. Ernst, and Olaf Schneider, *Analysis of acceleration strategies for restarted minimal residual methods*, Journal of Computational and Applied Mathematics, 123 (2000), pp. 261–292.

[11] Stanley C. Eisenstat, Howard C. Elman, and Martin H. Schultz, *Variational iterative methods for nonsymmetric systems of linear equations*, SIAM Journal on Numerical Analysis, 20 (1983), pp. 345–357.

[12] Mark Embree, *The tortoise and the hare restart GMRES*, SIAM Review, 45 (2003), pp. 259–266.

[13] Andreas Frommer, BiCGStab(l) *for families of shifted linear systems*, Computing, 70 (2003), pp. 87–109.

[14] Andreas Frommer and Uwe Glässner, *Restarted GMRES for shifted linear systems*, SIAM Journal on Scientific Computing, 19 (1998), pp. 15–26.

[15] Andreas Frommer, Stephan Güsken, Thomas Lippert, Bertold Nöckel, and Klaus Schilling, *Many masses on one stroke: Economic computation of quark propagators*, International Journal of Modern Physics C, 6 (1995), pp. 627–638.

[16] Beat Jegerlehner, *Krylov space solvers for sparse linear systems.*, Tech. Report IUHET-353, Indiana University, 1996.

[17] Wayne Joubert, *On the convergence behavior of the restarted GMRES algorithm for solving nonsymmetric linear systems*, Numerical Linear Algebra with Applications, 1 (1994), pp. 427–447.

[18] Misha E. Kilmer and Eric de Sturler, *Recycling subspace information for diffuse optical tomography*, SIAM Journal on Scientific Computing, 27 (2006), pp. 2140–2166.

[19] Sabrina Kirchner, *IDR-Verfahren zur Lösung von Familien geshifteter linearer Gleichungssysteme*, Diploma thesis, Bergische Universität Wuppertal, Department of Mathematics, Wuppertal, Germany, 2011.

[20] Björn Leder and Nemanja Bozovic, *Personal communication.* 2012.

[21] Ronald B. Morgan, *Implicitly restarted GMRES and Arnoldi methods for nonsymmetric systems of equations*, SIAM Journal on Matrix Analysis and Applications, 21 (2000), pp. 1112–1135.

[22] ———, *GMRES with deflated restarting*, SIAM Journal on Scientific Computing, 24 (2002), pp. 20–37.

[23] Michael L. Parks, Eric de Sturler, Greg Mackey, Duane D. Johnson, and Spandan Maiti, *Recycling Krylov subspaces for sequences of linear systems*, SIAM Journal on Scientific Computing, 28 (2006), pp. 1651–1674.

[24] Michael L. Parks, Rahul Sampath, and Phani K. V. V. Nukala, *Efficient simulation of large-scale 3d fracture networks via krylov subspace recycling*, (In Preparation).

[25] Yousef Saad, *Analysis of augmented Krylov subspace methods*, SIAM Journal on Matrix Analysis and Applications, 18 (1997), pp. 435–449.

[26] Yousef Saad, *Iterative methods for sparse linear systems*, SIAM, Philadelphia, Second ed., 2003.

[27] Yousef Saad and Martin H. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM Journal on Scientific and Statistical Computing, 7 (1986), pp. 856–869.

[28] Yousef. Saad, M. Yeung, J. Erhel, and Frèdèric Guyomarc'h, *A deflated version of the conjugate gradient algorithm*, SIAM Journal on Scientific Computing, 21 (2000), pp. 1909–1926. Iterative methods for solving systems of algebraic equations (Copper Mountain, CO, 1998).

[29] Valeria Simoncini, *On the convergence of restarted Krylov subspace methods*, SIAM Journal on Matrix Analysis and Applications, 22 (2000), pp. 430–452.

[30] ———, *Restarted full orthogonalization method for shifted linear systems*, BIT. Numerical Mathematics, 43 (2003), pp. 459–466.

[31] Valeria Simoncini and Daniel B. Szyld, *On the occurrence of superlinear convergence of exact and inexact Krylov subspace methods*, SIAM Review, 47 (2005), pp. 247–272.

[32] ———, *Recent computational developments in Krylov subspace methods for linear systems*, Numerical Linear Algebra with Applications, 14 (2007), pp. 1–59.

[33] ———, *Interpreting IDR as a Petrov-Galerkin method.*, SIAM Journal on Scientific Computing, 32 (2010), pp. 1898–1912.

[34] Charles F. Smith, Andrew F. Peterson, and Raj Mittra, *A conjugate gradient algorithm for treatment of multiple incident electromagnetic fields*, IEEE Transactions on Antennas and Propagation, 37 (1989), pp. 1490–1493.

[35] Peter Sonneveld and Martin B. van Gijzen, *IDR(s): a family of simple and fast algorithms for solving large nonsymmetric systems of linear equations*, SIAM Journal on Scientific Computing, 31 (2008), pp. 1035–1062.

[36] Gilbert W. Stewart and Ji-guang Sun, *Matrix Perturbation Theory*, Academic Press, San Diego, 1990.

[37] Eric de Sturler, *Nested Krylov methods based on GCR*, Journal of Computational and Applied Mathematics, 67 (1996), pp. 15–41.

[38] ———, *Truncation strategies for optimal Krylov subspace methods*, SIAM Journal on Numerical Analysis, 36 (1999), pp. 864–889.

[39] ———, *Convergence bounds for approximate invariant subspace recycling for sequences of linear systems*, in Program of the Householder Symposium XVIII on Numerical Linear Algebra, 2011, pp. 51–52.

[40] Henk A. van der Vorst and Kees Vuik, *The superlinear convergence behaviour of GMRES*, Journal of Computational and Applied Mathematics, 48 (1993), pp. 327–341.

[41] Shun Wang, Eric de Sturler, and Glaucio H. Paulino, *Large-scale topology optimization using preconditioned Krylov subspace methods with recycling*, International Journal for Numerical Methods in Engineering, 69 (2007), pp. 2441–2468.